FLIGHT SIMULATION TECHNOLOGIES CONFERENCE

A COLLECTION OF
TECHNICAL PAPERS

New Orleans, Louisiana
August 12 - 14, 1991

AIAA

A COLLECTION OF TECHNICAL PAPERS

Flight Simulation Technologies Conference

August 12-14, 1991 / New Orleans, Louisiana

Flight Simulation Technologies Conference

General Chairman
JACOB A. HOUCK
NASA Langley
Research Center

Technical Program Chairman
LOUIS H. KNOTTS
Calspan Corporation

Flight Simulation Technologies Conference
August 12-14, 1991 / New Orleans, LA

TABLE OF CONTENTS

N/A-Not Available   W-Withdrawn

FLIGHT TESTING OF THE CALSPAN VARIABLE STABILITY LEARJET 25 IN-FLIGHT SIMULATOR

Paul R. Deppe
Engineering Test Pilot
Arvin/Calspan Corporation
Buffalo, NY

Abstract

This paper describes the flight testing of the Calspan Variable Stability Learjet 25B, N102VS. This is the second variable stability Learjet which Calspan has developed. A description of the Learjet Model 25 and Variable Stability System (VSS) is presented, followed by a discussion of the scope of the tests. The results of the flight tests are presented, including basic aircraft systems tests, initial variable stability system engagement and checkout, envelope expansion, and gain configuration development. The potential applications and planned future improvements to the aircraft are also briefly discussed.

Background

Arvin/Calspan's variable stability Learjets are production Lear 20-series aircraft modified to serve as in-flight simulators. The first variable stability Learjet, N101VS, is a Lear 24D and has flown over 6,000 hours of variable stability operations in the past eight years. The low operating cost, high reliability and wide range of simulation capabilities of the Lear 24 has contributed significantly to test pilot training at the U. S. Navy and Air Force Test Pilot Schools and put this aircraft in great demand among government and civilian aeronautical research organizations. However, commitments to test pilot training have left little time for flight control systems research.

In late 1989 Arvin/Calspan purchased a Learjet model 25B for conversion into an in-flight simulator similar to N101VS. The aircraft, registration number N102VS, was intended to fill the demand for low-cost in-flight simulation both in the United States and abroad. The modifications were completed in March 1991, followed by a three-week flight test program at the Calspan Advanced Technology Center in Buffalo, NY. This paper presents the results of those flight tests.

Purpose

The purpose of this test was to flight test the basic aircraft systems and variable stability system in the variable stability Learjet 25, N102VS.

Fig. 1. Learjet 25B N102VS.

Description of Aircraft

The Learjet Model 25B is an all-metal, low wing, pressurized high performance executive jet with retractable landing gear. It is designed to carry two pilots and up to eight passengers. The Lear 25B is powered by two General Electric CJ610-8A single-rotor axial-flow turbojet engines which are mounted on pylons on each side of the aft fuselage. Each engine is rated at 3,100 pounds of thrust at sea level, standard day.

The Lear 25B flight control system is a mechanical reversible system with dual control wheels and rudder pedals in a side-by-side cockpit. Pitch trim is accomplished by changing the incidence of the horizontal stabilizer, while roll and yaw trim are accomplished with conventional trim tabs on the trailing edges of the left aileron and rudder, respectively. The longitudinal control system incorporates a bobweight and downspring designed to increase the apparent longitudinal static stability at low airspeeds. Spoilers and Fowler-type wing flaps are also incorporated and are electrically controlled and hydraulically operated.
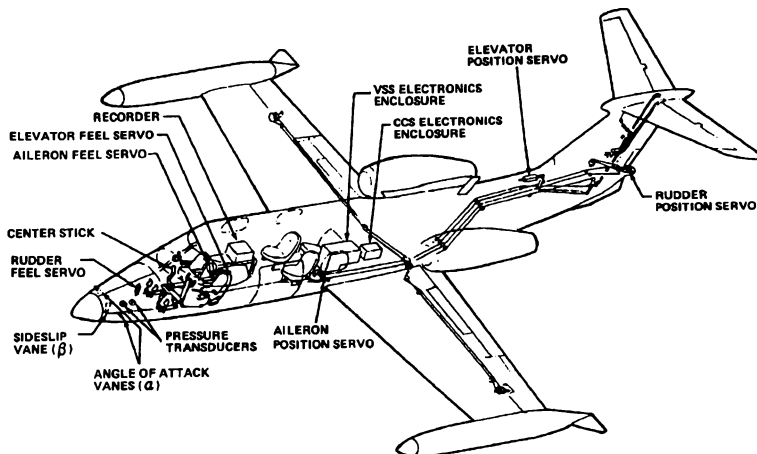
Fig. 2. Learjet N102VS Major Variable Stability System Components.

The test airplane, shown in Fig. 1, was a production Lear 25B and was representative of Lear 25B airplanes with the following exceptions:

1. The wing incorporates the Century III modifications which are designed to reduce the approach speed.

2. Fences, stall strips and boundary layer energizers were installed on the wings to improve takeoff and landing performance, increase aerodynamic stall warning, and improve roll control during stall or overspeed conditions.

3. The response-feedback variable stability system was instlled as described in subsequent paragraphs.

Description of VSS

Installation

The major modification to N102VS was the installation of a response-feedback, digitally-controlled analog variable-stability system. These extensive modifications required almost complete disassembly of the aircraft, including removal and reinstallation of the tail, engines, avionics, interior equipment, and all instrument panels. Major components of the variable stability system were located as shown in Fig. 2.

The major changes to the airplane were:

1. The copilot's control wheel and column were removed and a variable-feel, electro-hydraulic center stick was installed. The copilot's rudder pedals were disconnected from the aircraft system and connected to the variable-feel system as well.

2. The aileron control system was modified by the addition of a drum and cable arrangement for the attachment of the aileron servo actuator.

3. Elevator and rudder servos were installed in parallel with the normal control system, similar to autopilot servo installations.

4. The elevator bobweight and aileron-rudder interconnect were removed.

5. Panels for operating and monitoring the variable stability system were added to the instrument panel, center console and left side panel.

Operation

In a response feedback VSS, command and feedback signals are used to augment basic airplane stability characteristics in order to simulate the responses of other aircraft. The Lear 25 VSS is mechanized as shown in Fig. 3.

2
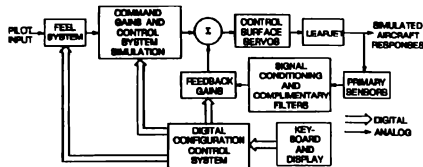
Fig. 3. Lear 25 Variable Stability System Block Diagram.

The variable-feel system is capable of simulating a wide range of stick and rudder pedal force/displacement gradients and nonlinear characteristics such as preload, freeplay and friction. Force or position command signals from the feel system are scaled and filtered as desired and summed with feedback signals to produce command signals to the surface actuators.

When the VSS is engaged, the hydraulic actuators move the control surfaces in response to commands generated by the VSS. The left seat controls are still mechanically connected to the surfaces, thereby allowing the safety pilot to monitor control surface activity. When the VSS is disengaged, the actuators are depressurized, allowing the control surfaces to be moved by the safety pilot's controls.

While engaged, the VSS monitors selected parameters and automatically disengages the system if preset levels are exceeded. The safety trip monitor includes the following parameters:

Surface and feel system rates
Surface hinge moments
Linear accelerations
Angles of attack and sideslip
Vertical tail loads
Hydraulic fluid level
Attitude gyro status
Configuration control system status

An emergency engage mode, called the Fly-By-Wire (FBW) mode is provided to allow the evaluation pilot to fly the aircraft in the event of safety pilot incapacitation. The FBW mode uses fixed command gains only to implement a Learjet-like aircraft optimized for landing and approach. All basic Learjet flight controls (including flaps, spoilers and brakes) are available except for nosewheel steering.

VSS gains are controlled by the microcomputer-based Configuration Control System (CCS). The CCS uses 64 multiplying digital-to-analog converters to set command and feedback gains, feel system parameters, and control system characteristics (time delay, filters, etc.). A list of the parameters which may be controlled is presented in Table I.

Table I. VSS Variable Gains.

$\delta_e/\Delta\alpha$	F_{ES} Preload	δ_e T. Delay
$\delta_e/\dot{\alpha}$	F_{ES} Friction	δ_e Lead
δ_e/q	F_{ES} Nonlinear Grad	δ_e Lag
$\delta_e/\dot{q}$	δ_{ES}/F_{ES} (Force)	δ_e Freeplay
$\delta_e/\Delta V$	F_{ES}/δ_{ES} (Position)	δ_e/F_{ES}
$\delta_e/\dot{V}$	$F_{ES}/\Delta V$	$F_{ES}/\dot{q}$
δ_e/δ_{ES}	$F_{ES}/\Delta N_z$	
δ_a/β	F_{AS} Preload	δ_a T. Delay
$\delta_a/\dot{\beta}$	F_{AS} Friction	δ_a Lead
δ_a/p	F_{AS} Nonlinear Grad	δ_a Lag
δ_a/r	δ_{AS}/F_{AS} (Force)	δ_a Freeplay
δ_a/δ_{AS}	F_{AS}/δ_{AS} (Position)	δ_a/F_{RP}
δ_a/F_{AS}	δ_a/δ_{RP}	
δ_r/β	F_{RP} Preload	δ_R T. Delay
$\delta_r/\dot{\beta}$	F_{RP} Friction	δ_R Lead
δ_r/p	F_{RP} Nonlinear Grad	δ_R Freeplay
δ_r/δ_{RP}	F_{RP}/δ_{RP} (Position)	δ_r/F_{AS}
δ_r/F_{RP}	δ_r/δ_{AS}	

The CCS stores 256 sets of gains (called "configurations"), 128 of which are pre-programmed and the remaining 128 of which may be stored in flight. The CCS control panel allows the pilot to call up any configuration and to display and change any gain without disengaging the VSS.

<u>Scope of Tests</u>

The flight tests were conducted during seven flights totalling 14.9 flight hours. Test configurations are defined in Table II.

Table II. Test Configurations.

CONFIGURATION	POWER SETTING	LANDING GEAR	FLAPS	SPOILERS
Cruise (CR)	TLF[1]	Up	Up	Retracted
Powered Approach (PA)	TNA[2]	Down	20°	Retracted

Notes: (1) TLF - Thrust for level flight at test airspeed.
(2) TNA - Thrust for normal approach (3° glide slope).

Test conditions are listed in Table III.

3

Table III.   Test Conditions.

FLIGHT NO.	TEST	ALTITUDE (ft MSL)	AIRSPEED (KIAS/MACH)
1	Basic aircraft systems and air data system.	0 - 41,000	0 - 275/0.70
2	Basic aircraft systems and envelope expansion.	0 - 45,000	0 - 360/0.82
3	FBW mode, safety trips, VSS mode, flutter tests, landings in FBW mode.	15,000	170 - 325
		23,000	170 - 275
4	Longitudinal command and feedback gain checkout	15,000	250
5	Lat-Dir command and feedback gain checkout	15,000	250
6	CCS configuration development	15,000	250
7	CCS configuration development, long-range navigation system checkout	0 - 41,000	0 - 300

The flight test program was originally planned to encompass fifteen flights over a one-month period. The schedule was compressed to seven flights in two weeks due to a commitment to the French Test Pilot School to be ready to conduct test pilot training in France in April 1991.

Instrumentation

Data were recorded on the onboard 24-channel AR700 digital tape recorder. For the first flight, the recorder was patched to record the signals required to check out the air data system and complementary filters. For subsequent envelope expansion flights, the recorder was patched to record safety of flight parameters such as accelerations, surface positions, and servo error signals.

Sixteen channels of data were also monitored in real time on the ground using Calspan's Flight Test Control and Telemetry Facility. The telemetry stream was recorded on digital magnetic tape as a backup to the airborne recorder.

Results

Basic Aircraft Systems Tests

Basic Learjet systems were checked during the first two test flights. The flight profiles were patterned after the Learjet factory production flight test cards with appropriate modifications for specific systems in the test aircraft. Although not tested on the first two flights, the VSS was installed

and operational to allow the pilots to use the FBW mode if necessary. As the mass properties of the flight control system had been changed by the installation of the VSS surface servos, the flight envelope was restricted for the first flight to reduce the chances of control surface flutter. Indicated airspeed was limited to 275 KIAS/0.70 IMN and normal acceleration was limited to +0.5 to +2.0 Gs.

The first flight was relatively uneventful except for several minor aircraft systems problems. A cabin pressurization problem prevented the aircraft from performing tests above FL300 so these tests were deferred until the second flight. All safety-of-flight-related discrepancies were corrected prior to the second flight.

On the second flight, the flight envelope was expanded to the limits of the production Learjet 25, and all remaining aircraft systems were checked. No abnormal control system vibrations were noted and the aircraft was cleared for testing with the VSS engaged.

Initial VSS Tests

The first VSS engage sequence was conducted in configuration CR at 15,000 ft MSL, 170 KIAS. These conditions were chosen so that in case of a surface servo hard-over failure, full control deflection would not result in exceeding any airplane limits. The VSS feedback gains were set to zero and the surface command gains were preset to nominal values. After engaging the feel system, the longitudinal surface servo was engaged and the evaluation pilot applied several test and trim inputs. Ground personnel monitored sensor signals, accelerations, and servo commands on strip charts driven from the real-time telemetry stream. The engage procedure was then repeated for the lateral and directional axes individually. All controls operated normally. The initial VSS tests were concluded with a check of the four manual disengage switches in the cockpit.

After verifying VSS operation at 170 KIAS, the VSS envelope was expanded to 325 KIAS in 25-knot increments. At each test point a series of control raps, doublets, and steps was performed to test for possible aeroservoelastic (ASE) oscillations. These tests were repeated at 23,000 ft MSL between 170 and 275 KIAS. At each "corner" of the envelope, a wind-up turn was performed, gradually increasing Nz until the automatic safety trips disengaged the system. No ASE problems were noted, the safety trip system worked properly, and the VSS system was cleared for the design flight envelope.

The FBW mode was tested in the same manner and at the same flight conditions as the normal VSS mode. Automatic safety trips, though normally inactive in this mode, were activated for these tests. Engage procedures were the same as in the initial VSS tests except that all three axes were engaged simultaneously. Again, no ASE problems were noted and the FBW system was cleared for the design flight envelope.

The first landings with the VSS engaged were conducted in configuration PA in the FBW mode, followed by landings in the VSS mode.

VSS Gain Envelope Expansion

VSS command path and feedback gains were tested in configuration CR at 15,000 ft MSL, 250 KIAS by gradually increasing the value of the gain and introducing a sharp control rap in the appropriate axis. When the gain reached the limit of adjustment or other limit, airspeed was increased to 325 KIAS in 25-knot increments and additional control raps were performed.

Due to the large number of combinations possible with 64 gains, each one adjustable in 1,000 steps, all combinations of gains could not be tested. All gains were exercised individually and in the following combinations:

1. δ_e/α and $\delta_e/\dot{\alpha}$

2. δ_e/α and δ_e/q

3. δ_r/β and $\delta_r/\dot{\beta}$

4. δ_r/β and δ_r/r

The range of short period frequencies and damping ratios tested is shown in Fig. 4.
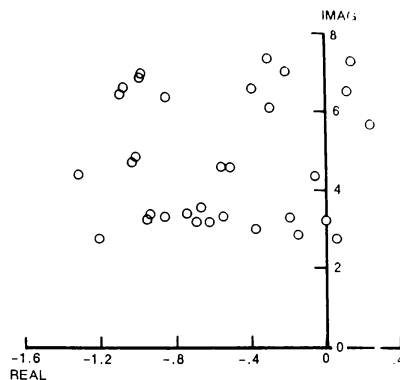
Fig. 4. Short Period Characteristics.

The range of dutch roll frequencies and damping ratios tested is shown in Fig. 5.
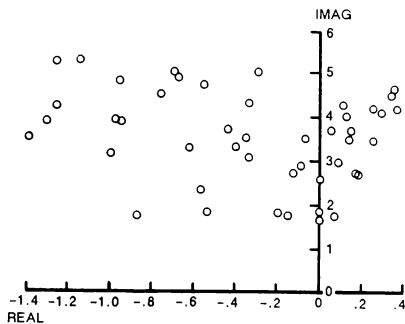
Fig. 5. Dutch Roll Characteristics.

5

The range of roll mode time constants tested is shown in Fig. 6.
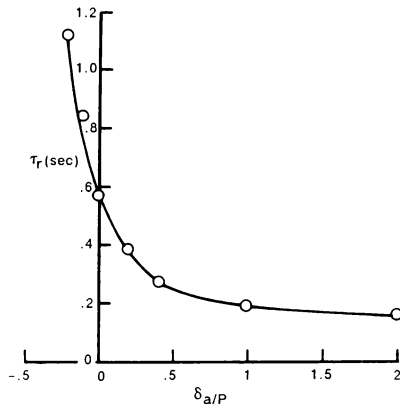
Fig. 6. Roll Mode Characteristics.

CCS Configuration Development

CCS configurations 0 through 127 are pre-programmed sets of gains used to demonstrate a wide range of aircraft flying qualities. For example, one configuration simulates a 4 rad/sec short period natural frequency, another a 6 rad/sec frequency, etc. These "demonstration" configurations were designed to match the flight characteristics presently in use on Calspan's first variable stability Learjet so that either aircraft could be used in the Navy and Air Force Test Pilot School syllabi.

Each configuration was evaluated in configuration CR at 15,000 ft, 250 KIAS, the nominal flight conditions for VSS flight demonstrations. After engaging the VSS, the CCS was used to adjust the gains as necessary to produce the desired static and dynamic responses for each configuration. Ground personnel monitored aircraft responses on strip charts driven from the real-time telemetry stream and provided recommendations for gain changes to the flight crew. The only gains which were significantly different from the Lear 24 were the pitch rate, yaw rate, angle-of-attack rate, and angle-of-sideslip rate feedback gains, which correlate to the longer fuselage and higher moments of inertia of the Lear 25. Even these gains, however, required adjustments of less than 10% from the Lear 24 values.

Summary

Flight testing of the second Calspan Variable-Stability Learjet, N102VS, has been completed and the aircraft is now operational. As of this writing the aircraft has operated at the U. S. Naval Test Pilot School, the French Test Pilot School (Istres), and the International Test Pilot School in Cranfield, U. K. No major problems have been encountered with the aircraft or VSS and the aircraft is expected to enjoy the same high reliability rates as Calspan's first variable stability Learjet.

Planned future enhancements include the installation of additional controllers in the evaluation cockpit. A variable-feel yoke (interchangeable with the center stick) will be used to simulate a wide variety of advanced transport and business-class aircraft, and a variable-feel sidestick will allow evaluation of military cockpit configurations.

A programmable general-purpose digital computer will be installed which may be used to simulate flight control systems, drive advanced cockpit displays, or prototype advanced navigation systems. The digital computer will incorporate a graphical user interface to allow rapid prototyping and automatic code generation from the block diagram design level.

The present three-degree-of-freedom capability of the Learjet 25 will be expanded to include direct-lift flaps and servoed throttles. The additional control power provided by these enhancements in conjunction with the digital computer will allow the Learjet 25 to simulate gust alleviation schemes, adaptive flight controls, and the flight characteristics of modern highly-augmented aircraft.

Calspan's Learjet 25 In-Flight Simulator provides the flight controls research and development community with a low cost, quick-response platform with which to evaluate advanced flight control system concepts and ideas.

IN-FLIGHT SIMULATION AT THE NASA DRYDEN FLIGHT RESEARCH FACILITY

Mary F. Shafer*
NASA Dryden Flight Research Facility
Edwards, California

Abstract

Since the late 1950's the National Aeronautics and Space Administration's Dryden Flight Research Facility has found in-flight simulation to be an invaluable tool. In-flight simulation has been used to address a wide variety of flying qualities questions, including low-lift-to-drag ratio approach characteristics for vehicles like the the X-15, the lifting bodies, and the Space Shuttle; the effects of time delays on controllability of aircraft with digital flight-control systems, the causes and cures of pilot-induced oscillation in a variety of aircraft, and flight-control systems for such diverse aircraft as the X-15 and the X-29. In-flight simulation has also been used to anticipate problems and to avoid them and to solve problems once they appear.

This paper presents an account of the in-flight simulation at the Dryden Flight Research Facility and some discussion. An extensive bibliography is included.

Nomenclature

C^*	blended normal acceleration, pitch rate, and pitch acceleration
DFBW	digital fly-by-wire
DFRF	Dryden Flight Research Facility, Edwards, CA
FCS	flight-control system
GPAS	General Purpose Airborne Simulator
HUD	head-up display
L/D	lift-to-drag ratio
LLRV	Lunar Landing Research Vehicle
NASA	National Aeronautics and Space Administration
NASP	National AeroSpace Plane
PIO	pilot-induced oscillation
RAV	remotely augmented vehicle
RPRV	remotely piloted research vehicle
SST	Supersonic Transport
TIFS	Total In-Flight Simulator
USAF	United States Air Force
VSA	variable-stability aircraft
$1/\tau_{\theta_2}$	high-frequency pitch attitude zero
$\dot{\beta}$	sideslip rate, deg/sec
$\omega_{n_{sp}}$	undamped natural frequency of the short period mode, rad/sec

Introduction

Before flying an experimental aircraft it is always desirable to consider the flying qualities of the vehicle. If the new vehicle is similar to an existing aircraft, this may provide an idea of the flying qualities of the new vehicle. New aircraft of unusual configuration or flight envelope, however, require special handling.

Ground-based simulation is a good tool to use for an initial examination of the flying qualities, but ground-based simulators are deficient when reproducing visual or motion cues. They are suitable for many regions in the envelope, like cruise, but more demanding tasks, such as precision landings, frequently cannot be simulated well enough to provide complete confidence.

In-flight simulation does not have the same limitations as ground-based simulation. Visual cues are identical with those in the subject aircraft and motion cues, if the simulation is modeled correctly, also match those of the subject aircraft. In-flight simulation is also better at exposing deficiencies like proneness to pilot-induced oscillation (PIO). In fixed-base simulations, PIOs are not often seen, no matter how deficient the aircraft and its flight-control system (FCS), unless unusual, unrepresentative tasks are used. During in-flight simulation, these PIOs occur more readily.

There are two roles for in-flight simulation. The more difficult role is the examination of the dynamic response

*Aerospace Engineer. Associate Fellow, AIAA.

of an aircraft. Simulating the dynamic response (natural frequency and damping and the phasing between them, for example) of the subject aircraft requires modifying the dynamic response of the simulation aircraft. The variable stability aircraft used for dynamic simulation are the aircraft most often thought of when considering in-flight simulation.

The other role of in-flight simulation is performance simulation. This is the use of a similar aircraft to explore various performance characteristics which are not highly dynamic. An example of performance simulation is the use of an F-104 Starfighter in a low-lift-to-drag ratio (L/D) configuration to simulate the X-15 aircraft in approach and landing. No modification to the F-104 aircraft was required for this simulation, because the F-104 can easily be configured with low L/D.

In-flight simulation is more difficult, more time-consuming, and frequently more expensive than ground-based simulation and is reserved for those portions of the flight regime that cannot be adequately evaluated on the ground. It is not a cure all, as the simulation is only as good as the understanding of the characteristics of the simulated aircraft. The limitations of the simulator aircraft also limit the fidelity of the simulation.

The mission of the National Aeronautics and Space Administration's Dryden Flight Research Facility (NASA DFRF) is the study and flight test of a variety of unconventional and experimental fixed-wing aircraft. Dryden has used in-flight simulation to support this mission since the late 1950's. The first simulation was a generic study into the approach and landing of low-L/D aircraft using an F-104. The most recent was a 1990 inquiry into the visibility requirements in the approach and landing of a hypersonic vehicle using an F-104 aircraft.

Between these two simulations there have been a wide variety of simulation programs, using both dynamic and performance simulators to simulate such diverse subject aircraft as the X-15, the lifting bodies, the X-20 DynaSoar, and the X-29. Extensive inquiries into a variety of flying qualities topics have also been made. In keeping with the limitations of in-flight simulation, only pertinent portions of the flight regimes of the various aircraft have been studied.

This paper, a history of in-flight simulation at DFRF, describes the dynamic flight simulators and many of the performance simulators and presents a brief chronology of in-flight simulation here. The summary discusses a number of common threads in the history. An extensive bibliography is provided for further information.

Description of Simulator Aircraft

There are two types of in-flight simulation, dynamic and performance, and, hence, two types of simulators. The dynamic simulator aircraft are extensively modified because control of the dynamic response is difficult. Com-

puters control the actual response, completely overpowering the natural response of the aircraft. This complexity also means that these simulators provide the most information about flying qualities because they can be made to fly like different aircraft. In addition, the more recent of these variable-stability aircraft can be used to assess a variety of FCSs because the aircraft already have powerful and flexible flight-control computers.

The aircraft used for the in-flight simulation of the performance of the subject aircraft are much simpler. Typically, modifications are small changes to existing structures–a bigger speed brake, for example, to match the L/D of the subject aircraft better. These performance simulators are frequently used to provide information about the feasibility of a flight task, to provide qualitative information about a generic class of aircraft, or to establish piloting techniques. At the DFRF, the performance simulators were frequently support aircraft, pressed into duty when the need arose. This is particularly conspicuous in some of the visibility studies, where card or plastic was used to block the windows of standard support aircraft.

Performance simulation is less versatile than dynamic simulation because it is limited by the performance of the simulator aircraft. For example, the unmodified F-104 aircraft was not suitable for simulating the X-15 in any other flight regime, but it was an excellent simulator in the pattern.

Dynamic Simulators

Variable-Stability F-100C Super Sabre. The NASA F-100C Super Sabre, (Fig. 1) a single-engine swept-wing supersonic fighter, was modified by the Ames Research Center as a variable-stability research vehicle that provided variation of parameters around all three axes.[1,2] An analog fly-by-wire system was used in all three axes, although the pitch axis had safety trips installed because of the run-away potential of the all-moving horizontal tail.
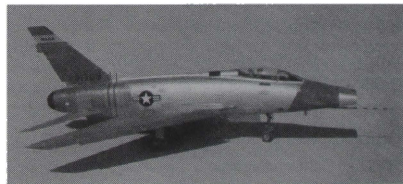
EC62 145
Fig. 1 The NASA F-100 Super Sabre aircraft.

NT-33A Variable Stability Aircraft. The United States Air Force NT-33A variable stability aircraft (VSA) (Fig. 2) is an extensively modified T-33A Shooting Star jet trainer.[3] The most conspicuous modification is the enlarged nose section that provides more room for electronics. The front seat, where the evaluation pilot sits, has a standard center stick or

side stick and rudder pedal arrangement. The standard front seat control system has been replaced by a full authority fly-by-wire FCS and a variable response artificial feel system. The safety pilot sits in the rear seat to program the configuration characteristics.

EC87 0126-7

Fig. 2 The USAF NT-33A variable stability aircraft.

The NT-33A aircraft has independent control of three-degrees-of-freedom for in-flight simulation. The simulation technique uses a response feedback methodology with three moment controllers of the vehicle (elevator, aileron, and rudder) as the simulation effectors. At one time the NT-33A had drag modulation, using drag petals at the wingtips, but this feature was removed following a structural failure.

The General Purpose Airborne Simulator. The NASA General Purpose Airborne Simulator (GPAS) (Fig. 3) was a modified Jetstar, an executive transport airplane. The original modifications made the GPAS a four-axis simulator (pitch, roll, yaw, and thrust force along the longitudinal axis) with a model-following variable stability system.[4,5] Direct lift control and direct side force were eventually added. The evaluation pilot sat in the left seat, which had a special set of transport-airplane-type controls and displays. This simulator exhibited extraordinarily good model following and had remarkable fidelity.[6] Werner von Braun was taken for a demonstration flight early in the career of the GPAS. Impressed, he described it as a "dial-a-plane," the first known use of this phrase.[6]
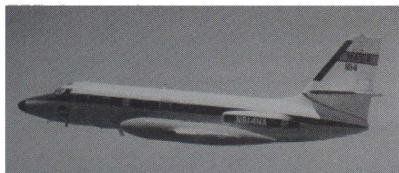
ECN 9718

Fig. 3 The NASA General Purpose Airborne Simulator aircraft.

The Total In-Flight Simulator. The USAF Total In-Flight Simulator (TIFS) is a highly modified C-131 aircraft configured as a six-degree-of-freedom simulator (Fig. 4). It has a separate evaluation cockpit forward and below the nor-

mal C-131 cockpit. The six-degrees-of-freedom are independently controlled by use of the elevator, aileron, rudder, throttle, direct lift flap, and side force surfaces. This side force surface is a large vertical surface mounted at mid-span of the wing. Longitudinal and lateral–directional model-following systems provide the evaluation pilot with motion and visual cues representative of the simulated aircraft. The evaluation cockpit can be modified with appropriate controls and displays and can accommodate a co-pilot. The TIFS can simulate turbulence and crosswinds or cancel an actual crosswind.

EC84 31731

Fig. 4 The USAF Total In-Flight Simulator aircraft.

The F-8 Digital Fly-By-Wire Aircraft. The NASA F-8 digital fly-by-wire (DFBW) was an F-8C Crusader, a single-engine, single-seat supersonic fighter (Fig. 5), with a full-authority digital fly-by-wire FCS.[7] The control system was designed so parameters such as time delays and control system gains could be entered from the cockpit in flight.
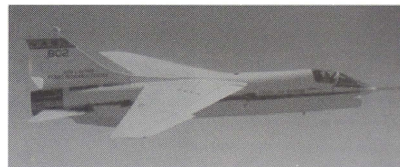
ECN 6982

Fig. 5 The NASA F-8 digital fly-by-wire aircraft.

The aircraft was also capable of accepting control-surface commands from a ground-based computer when in the remotely augmented vehicle (RAV) mode.[8–10] Using this feature, experimental control laws could be programmed in the ground-based computer, giving a special flexibility to simulation programs and keeping the evaluation pilot from knowing what configuration was being flown.

The Variable-Stability Learjet. The Calspan variable-stability Learjet (Fig. 6) is an executive transport aircraft, modified as a three-axis simulator with a response feedback flight-control system.[11] The evaluation pilot sits in the right seat, which is equipped with a center and a side stick which are, like the rudder pedals, driven by the variable feel system.

EC86 33565-3

Fig. 6 The Calspan variable-stability Learjet.

This aircraft was originally designed as a training tool for the Air Force and Navy Test Pilot Schools, but has been used by DFRF for flying qualities research.

Performance Simulators

The aircraft used in performance simulators are not extensively modified. Most of these aircraft were used for support at DFRF.

The F-102A Delta Dagger. The NASA F-102A Delta Dagger was a single-engine supersonic delta-wing interceptor aircraft (Fig. 7) that could be configured as a low-L/D aircraft in the power approach configuration.[2,12]

E-2551

Fig. 7 The NASA F-102A Delta Dagger aircraft.

The F-102A Delta Dagger was used for pilot proficiency, chase, and research studies. It was modified with a larger speed brake for certain low-L/D aircraft studies.

The F-104 Starfighter. The NASA F-104 Starfighter is a single-engine, Mach 2 aircraft with a small, straight wing and a T-tail.[2] The wing area is less than 200 ft^2 and the weight is approximately 24,000 lb, so it has a fairly high wingloading.[13] These F-104 Starfighters were used for pilot proficiency, chase, and as testbeds for a variety of experiments. The F-104B and TF-104G (Fig. 8), both two-seat Starfighter aircraft, were used in restricted visibility studies. Another Starfighter, the YF-104A, was modified with a reaction control system.
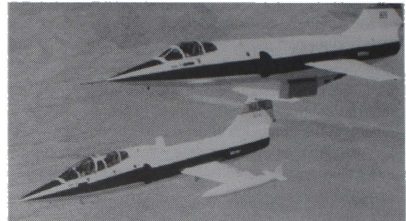
EC88-0151-001

Fig. 8 The NASA TF-104G Starfighter aircraft, lower left.

The F5D Skylancer. The NASA F5D Skylancer aircraft (Fig. 9) was designed as a carrier-based short range interceptor fighter.[14] It was a tailless single-engine aircraft with a swept back wing of extremely low aspect ratio; the planform resembling the proposed DynaSoar vehicle and some Supersonic Transport (SST) configurations. Enlarged speed brakes were used in a lifting body approach and landing study.

Fig. 9 The NASA F5D Skylancer aircraft.

The A-5A Vigilante. The twin-engine supersonic strategic bomber A-5A Vigilante (Fig. 10), operated by NASA, had a high wing, a rolling tail, and a slab fin.[14] The low-aspect-ratio swept back wing had no ailerons; blown flaps were used for low speeds and spoilers and rolling tail for high speeds. The aircraft also had variable-geometry intakes. This aircraft was borrowed from the U. S. Navy for use in the SST approach control studies.
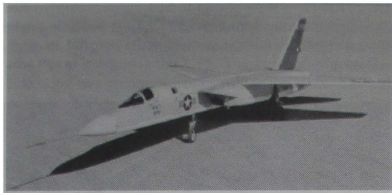
ECN 318

Fig. 10 The NASA A-5A Vigilante aircraft.

The NB-52B Stratofortress. The NASA NB-52B (Fig. 11) is a modified B-52B Stratofortress, a strategic bomber with a high, swept wing and eight engines.[2] This aircraft was modified to carry and launch the X-15. It has an inboard pylon on the right wing and a large notch in the inboard flap. Dryden acquired this airplane in 1959 and it is still in use.
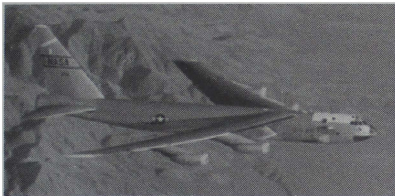
EC89 0345-41

Fig. 11 The NASA NB-52B Stratofortress aircraft.

The F-111A. The F-111A (Fig. 12) is a supersonic sweep-wing, twin-engine tactical bomber. The aircraft belonged to the USAF and was flown by NASA and air force pilots in support of the shuttle program.

ECN 2092

Fig. 12 An F-111A aircraft.

The CV-990. The NASA CV-990 (Fig. 13) was a four-engine transport aircraft that was used in several transport flying qualities investigations in the 1960's. This aircraft was then converted to an airborne observatory by NASA.

E-19753

Fig. 13 The NASA CV-990 aircraft.

The PA-30 Twin Comanche. The NASA PA-30 (Fig. 14) is an extensively modified PA-30 Twin Comanche, a twin-engine, low-wing, four-seat general aviation airplane. The modifications include a complete flight-test instrumentation system and an uplink–downlink system for telemetering pilot commands and aircraft response, for the emulation of remotely piloted research vehicles.[10] This airplane was acquired by DFRF in 1967 and is still in use.

ECN 2846

Fig. 14 The NASA PA-30 Twin Comanche aircraft.

The YF-12 Blackbird. The NASA YF-12 Blackbird (Fig. 15) was a twin-engine, Mach-3 interceptor aircraft. Two models, the YF-12A and the YF-12C (visibly differing mainly by the length of the chine), were used for supersonic research in propulsion, structures, and aerodynamic heating.[15] These airplanes were operated at DFRF from 1969 to 1979.

EC73 3685

Fig. 15 A YF-12 aircraft.

The F-15 Eagle. The NASA F-15 Eagle (Fig. 16) is a twin-engine, Mach-2 air superiority fighter. This aircraft, used in propulsion research, has an advanced digital engine control system.

11

E-19956

Fig. 18 The NASA T-33A aircraft.

F-104 Investigations of Approach and Landing Visibility

The F-104 aircraft have been used for many investigations into visibility requirements for approach and landing for low-L/D aircraft.[28] The first, in the early 1960's, used an F-104B aircraft with an indirect viewing system that had two wide-angle overlapping periscopes with stereoscopic vision, for conventional and low-L/D landings.[29] The periscopes were mounted on the canopy bow between the front and rear cockpits (Fig. 19) and the image was shown to the evaluation pilot in the rear seat. This system showed safe and acceptable performance in all phases of daylight flight. When the horizon was in the field of view, aircraft attitude sensing with the optics was satisfactory about all axes except pitch attitude in climbing flight. This degraded pitch-attitude sensing was caused by the poor resolution at the bottom of the field and the lack of view to the sides. However, this system had such large light loss and degraded resolution that it was not usable for night operations. It was also found that more view directly to the side was needed to perform circling approaches.
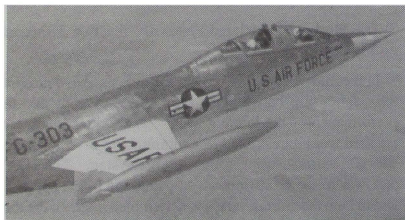
ECN 1297

Fig. 19 The NASA F-104B Starfighter aircraft with periscopes mounted on canopy bow.

The second study, with the same setup, examined the use of the stereoscopic periscope system in lifting body approaches and landings.[30] Three approach techniques (circling approach, straight-in approach, and a three-turn multiple-aim-point approach) had been proposed for lifting body approaches. The previous F-104B study had determined that the circling approach required side vision which the periscope system did not provide, so the two approach techniques requiring only forward vision were added to the assessment.

The previous F-104B program had left some doubt about the system's suitability for low-L/D approaches and landings because of the effects of exaggerated stereopsis at or near the ground. To solve this problem, a radar altimeter was also installed and pressure altitude, radar altitude, radar altitude rate, and indicated airspeed were inserted into the field of view of one of the periscopes. However, this early attempt at a head-up display (HUD) was unsuccessful, as the pilots found the information unreadable or unusable. Interestingly enough, pilots, with their excellent uncorrected vision, found this periscope system tiring and difficult to use while non-pilots who wore glasses did not have such problems. As in the study of conventional approaches and landings, the optical system provided adequate visual information for the flare and landing tasks and landing performance characteristics comparable to those obtained with normal vision. The exaggerated stereopsis played only a minimal roll in the high-speed landings, compared to the slower landings in the first study of this system.

The third F-104 limited visibility study, flown in the 1960's, involved masking the forward view, so the pilot had to rely on the field of view from side windows to land.[31] An appreciable amount of the forward field of view could be obscured before the landing performance suffered markedly.

In 1990, the fourth study used stencil board to mask the front cockpit field of view of the TF-104G.[28] This technique was also used in the third study. A number of windows, selected to match those proposed for the National AeroSpace Plane (NASP), were examined using straight-in approaches. In agreement with the earlier results, it was found that the pilot could land the plane with a fairly limited field of view. Unlike the earlier studies no circling approaches were examined, since it was assumed that some type of external guidance would deliver the airplane to the high-key position.

This TF-104G is currently being measured and instrumented for the installation of a folded-mirror optical viewing system which has been proposed for the NASP. This monoptic system for low-L/D approaches will be tested in the same manner as was the stereoptic system, with low-L/D approaches and precision landings.

NT-33A Simulation of M2-F2 Pilot Induced Oscillation

In 1965 the NT-33A aircraft was used to examine lateral–directional handling qualities of a variety of flight characteristics for the reentry mission.[32] One set of configurations matched the M2-F2 lifting body (Fig. 20) being tested at DFRF at the time. This simulation program found a coupled roll-spiral PIO (or lateral phugoid) which later manifested itself in the M2-F2.[21,33] The M2-F2 PIO was anticipated because it had been seen in the NT-33A simulation. This coupled roll-spiral PIO had been encountered in up-and-away

flight twice and had posed no problem. When this PIO was encountered on final approach it was quite severe and led to a serious accident.
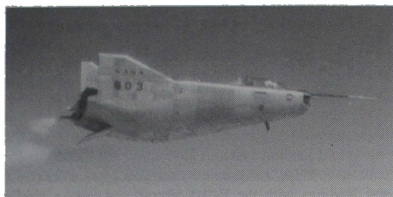
Fig. 20 The NASA M2-F2 lifting body.

Lunar Lander Research Vehicle

The Lunar Lander Research Vehicles (LLRVs) (Fig. 21), a program of the mid-1960's, were initially procured to examine the problems associated with lunar landing.[21,34,35] Lift and attitude control rockets were used during the landing simulations but the jet engine of the vehicle was used to lift and translate the craft to the simulation starting point. This led unavoidably to the examination of low dynamic pressure vertical take-off and landing flight. This jet engine was also used to counter 5/6 of the weight of the vehicle, simulating the lunar gravitational acceleration. The variable-stability control system permitted the examination of attitude command and of rate command with on–off control acceleration and proportional acceleration. Pilots discovered that attitude command was easier to fly than rate command and that satisfactory control was more easily achieved in rate command with on–off control acceleration than with proportional control.
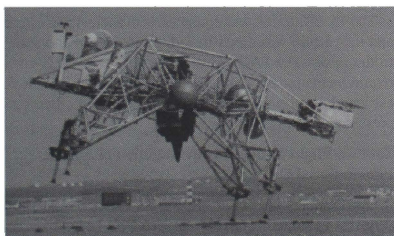
Fig. 21 The NASA Lunar Lander Research Vehicle.

The visual, motion, and audio cues made the simulation highly effective. The LLRVs were so successful at simulating lunar landings that they were transferred to the space program[21] and used for astronaut training, renamed Lunar Lander Training Vehicles, type A or LLTV-A. Three more

derivative vehicles, the LLTV-Bs, were later acquired by the space program.

General Purpose Airborne Simulator Simulation of Supersonic Cruise

The GPAS was programmed to simulate the Mach-3 XB-70 aircraft (Fig. 22) as part of the initial testing of the aircraft in the mid-1960's.[36,37] After this testing, the simulation was used as a pilot training tool in the XB-70 program and was also proposed for evaluation of the cruise regime of proposed SSTs.[38] The F100C aircraft was also used to study SST flying qualities.[21] The F5D Skylancer was used to establish minimum speed criteria for the proposed SST.[15]

Fig. 22 The XB-70 aircraft.

General Purpose Airborne Simulator Investigation of Motion and Visual Cues

An interesting part of the mid-1960's initial testing of the GPAS system was a study of motion and visual cues.[37] The effects of mismatched cues on observed handling qualities were studied by varying yaw rate and lateral acceleration at the pilot's location, while keeping constant the lateral–directional dynamics displayed on the pilot's instruments. This experiment showed pilot sensitivity to directional motion cues to be different for the simulation of two XB-70 flight conditions. Motion cue effects were determined using consecutive evaluation of moving-and fixed-base configurations in flight.

The second area investigated in this study was the measurement and description of simulation fidelity. In-flight frequency response measurements of the model-following system were taken to examine model-following fidelity for directly matched variables such as sideslip and roll rate as well as uncontrolled parameters such as lateral acceleration.

General Purpose Airborne Simulator Investigation of Roll Handling in Cruise and on Approach

The GPAS was used to evaluate roll handling for transport aircraft in both cruise and approach in the mid-1960's.[40,41] In cruise, maximum roll-control angular acceleration, maximum available roll rate, roll time constant, and bank-angle change in a given time were all found to be effective roll-criteria parameters and the criteria developed in this program agreed well with previously proposed roll criteria. In approach, maximum roll rate, roll time constant, and wheel characteristics were varied.

General Purpose Airborne Simulator Simulation of the HL-10 Lifting Body

In 1967 the GPAS was used to investigate the longitudinal flying qualities of the HL-10 lifting body.[42] Two flights were flown, but the simulation was not entirely satisfactory because of limitations in the closed-loop response of the GPAS.[6]

General Purpose Airborne Simulator Investigation of Ride Qualities

In the early 1970's the GPAS was used to investigate ride qualities, particularly in turbulence. In the first study, subjects (naive non-pilots recruited from the DFRF support staff and junior engineers) evaluated the ride quality and any motion sickness symptoms that manifested themselves. This information was compared to the dynamic data collected during the various runs. From this data a number of ride quality rating models were proposed. The assessments were also compared to assessments made by a number of passengers on scheduled airliners.

In 1973-74 several ride smoothing flight-control systems (basic, command augmentation, and rate feedback) were evaluated in turbulence. These flight-control systems were designed to maintain good flying qualites while smoothing the ride, to the advantage of pilots and passengers.[42] In the longitudinal axis command, augmentation systems reduced the normal acceleration response and the flightpath angle disturbances, compared to the basic and rate feedback systems, by greatly reducing the phugoid response. However, the calculated ride quality ratings showed only small improvements.

In the lateral–directional axes, significant reductions in roll rate, yaw rate, and lateral acceleration responses to turbulence were seen with a rate feedback system. The command augmentation systems were no better at reducing these responses; however, they did provide a significant reduction in bank angle and heading angle disturbances, which are of interest from a piloting standpoint. Some of the ride quality rating models indicated that these improvements modified the ride greatly while others showed no effects, depending on how greatly the lateral–directional variables were believed to affect the ride.

It was during a flight in support of this mission that the GPAS suffered an over-g condition and was retired. However, after new wings were installed, this aircraft was used as a testbed for a variety of experiments, including propulsion and boundary-layer control.

Shuttle Simulation Using Large, Low-L/D Vehicles

In support of the Space Shuttle Program, simulations of the shuttle using large, low-L/D vehicles, were undertaken in the late 1960's and 1970 using the NB-52B (Fig. 11), an F-111A (Fig. 12), and a CV-990 (Fig. 13).[43,44] These large aircraft were configured for low L/D (the CV-990 had L/Ds of approximately 5 to 8, the NB-52B had L/Ds of about 3.3 to 8, and the F-111A had L/Ds from about 6.6 with the wings at 26° to about 3.7 with the wings at 72.5° and the gear down) and the engines shut down or throttled back sufficiently to produce power for necessary systems only.

The NB-52B and CV-990 aircraft were initially used to evaluate the feasibility of landing such vehicles. Once it was determined that large, low-L/D aircraft could be landed visually, the programs were expanded to examine instrument flight rules (IFR) approaches and landings with the NB-52B, instrument landing system (ILS) approaches and landings with the F-111A, and ground-controlled approaches (GCA) and landings with the F-104 aircraft. Again, a circling approach was found to provide the best energy management and control of the touchdown point. A YF-12 aircraft (Fig. 15) was also used as part of this effort to develop baseline flying qualities data for large, low-L/D aircraft in the approach and landing.

PA-30 Emulation of Remotely Piloted Research Vehicles

The PA-30 aircraft (Fig. 14) was used in the early 1970's in a remotely piloted mode[8,45] to practice piloting techniques for a variety of unmanned remotely piloted research vehicles (RPRV), including the 3/8-scale F-15 RPRV, an unpowered model used in spin testing; the Drone for AeroStructural Testing (DAST) aircraft, a modified Firebee drone used to examine aeroelasticity; and the Highly Maneuverable Aircraft Technology (HiMAT) aircraft, an aerodynamically-advanced supersonic RPRV.

The PA-30, a low-wing, twin-engine general aviation airplane, provided training and currency for the exacting task of landing the RPRVs, and some currency in the ground cockpit. In addition, a variety of cameras and displays were tested to determine effective ways of presenting information to the pilot of a remotely piloted aircraft.

The PA-30 aircraft was equipped with a television camera and the picture was down-linked to the ground and shown to the pilot. The PA-30 was later used to research visual requirements for the remote piloting task, with various focal lengths and fields of view being examined. Stereoptic presentations were also examined.

Total In-Flight Simulator Investigation of Shuttle Pilot-Induced Oscillation

On October 26, 1977 the Space Shuttle Enterprise (Fig. 23) exhibited a fully-developed PIO in both the roll and pitch axes during a landing on the paved runway during the approach and landing test program. As a result, in 1978 the Total In-Flight Simulator (TIFS) aircraft was used in a simulation program to discover and confirm the reasons for this PIO.[46]

Fig. 23 The Space Shuttle Enterprise about to touch down on the paved runway.

Analysis indicated that PIO was caused by several factors, among them time delay in the FCS and the position of the pilot relative to the center of rotation.[46] The pilot's position masked the normal motion cues, since the pilot was somewhat behind the center of rotation. Surface rate limiting also contributed to the apparent time delay. This simulation confirmed the effects of these factors.

F-8 Digital-Fly-By-Wire Evaluation of Effects of Time Delay on Handling Qualities

Immediately following the TIFS investigation of the shuttle PIO, the F-8 DFBW was used in a test program to study the effects of time delays in a digital control system like that of the shuttle and to provide more insight into the shuttle approach and landing experience.[47,48] Transport delays were inserted into the roll and pitch axes and evaluated with formation flying and precision landing approaches (straight in and offset) at idle power, simulating the low-L/D approach typical of the Space Shuttle. In the pitch axis three different control modes were examined; stability augmentation, command augmentation, and no augmentation. The addition of time delay markedly affected the pilot's ability to control the airplane, to the point that the pilot scraped the tail of the plane on the runway during one go-around.

Formation flight was much less sensitive to the effects of time delay than was the approach task. Offset landing (where the pilot could not set up the approach but had to fly the plane more aggressively) was approximately twice as sensitive to time delay as was the straight-in approach. Furthermore, the ratings in pitch were most strongly affected by the task and were only slightly affected by changes in control system augmentation mode.

Total In-Flight Simulator Investigation of Shuttle Pilot-Induced Oscillation Suppressor Filters

Further investigation of the shuttle PIO led to the design of two candidate PIO suppression filters to control the problem. Flown in 1979, this TIFS investigation examined two PIO suppression filters that were proposed as an addition

to the shuttle FCS.[49] In addition, this program also examined some other modifications to the shuttle FCS, including feedforward of the pitch command and normal acceleration feedback. The effects of moving the pilot forward 100 ft were also investigated, although this was not proposed as a solution to the PIO problem. One of the two PIO suppressors evaluated in the TIFS program was implemented in the shuttle FCS prior to its first flight.[50]

Pilot-Induced Oscillation Suppression Filter Assessment with the F-8 Digital Fly-By-Wire

The Space Shuttle PIO, caused in part by excessive time delay and the success of the PIO suppression filters devised to alleviate the problem, created an interest in the usefulness of PIO suppression filters in more conventional aircraft. In 1980 there was a program to evaluate the same types of filters in more conventional fighter-type aircraft using the F-8 DFBW.[7] As previously described, the F-8 DFBW aircraft had already been used to evaluate the effects of time delay on digital FCSs so that the only addition required was the PIO suppression filters. The same two types of filters were examined, with a variety of breakpoints and filter slopes. The basic F-8 DFBW configuration was a good airplane with little time delay. Either a pure time delay or a first order lag was added to the FCS. The latter was used to simulate the cascading of filters in a poorly designed control system.

To provoke any possible PIO, two high-gain tasks were used. The first task, close-trail formation, involved flying just behind and below the F-104 chase plane. Pilots found that this task was somewhat artificial and not well defined. As a result of this assessment a more demanding task, probe-and-drogue refueling, was used in the second phase of the program. However, the results for the two tasks did not vary much.

The PIO suppression filters suppressed the PIOs in the configurations with added transport delay. They did not, however, help with the configurations with the first order lag.

NT-33A Pilot-Induced Oscillation Suppression Filters

In 1981 a simulation program was flown in the NT-33A aircraft to investigate PIO suppression filters in fighter-type aircraft.[7,51,52] A basically good configuration was selected as the baseline. To this baseline were added either time delay or lag pre-filtering in the longitudinal axis, similar to the F-8 DFBW PIO suppression filter study, and the same two PIO suppression filters were examined. In this study, the task was a precision offset landing.

The results of this and the F-8 DFBW experiments matched the shuttle program results, indicating that PIO suppression filters worked well for fighter-type aircraft as well. The PIO suppression filter greatly reduced PIOs, even with excessive time delay that led to serious PIOs in configurations without the filter. Already good flying qualities were not degraded by the filters. However, in the NT-33A

study, as in the F-8 DFBW studies, the filters made configurations with lag pre-filtering worse, indicating that poor system design could not be compensated for with the PIO suppression filters.

F-8 Digital Fly-By-Wire Investigation of Nonlinear Control Algorithms

In the early 1980's the F-8 DFBW was used to investigate active, nonlinear flight-control techniques and handling qualities in a cooperative program with the Royal Aircraft Establishment.[53] The evaluation was accomplished using the RAV mode.

The purpose of the study was two-fold, with the first goal being to establish whether a variable-gain controller could offer improved control performance over a linear baseline pitch-rate command system and whether any adverse handling problems would be introduced by the rapidly varying gain. The second goal was to investigate the effects of a nonlinear command pre-filter. The nonlinear pre-filter was designed to provide a small overshoot on the pitch rate and a relatively slow buildup of normal acceleration for small commands and to increase the pitch-rate overshoot and normal acceleration response for large commands. This was accomplished by varying the lead time constant of the pre-filter.

Distant tracking and close tracking were the two typical fighter tasks evaluated. The nonlinear pitch-rate command system worked well in the distant-tracking task; however, it was discovered that different responses are preferred for the two different tasks. Low-overshoot pitch-rate responses are preferred in the distant-tracking task and high-overshoot pitch-rate responses are preferred in the close-tracking task.

Nothing conclusive was learned about the variable adaptive, lead pre-filter time constant because the range of pre-filter time constants was not sufficiently related to the augmented dynamics. The F-8 DFBW aircraft, with its versatile FCS, was also used at this time in a brief, undocumented study of roll mode time constant and roll ratcheting.

Total In-Flight Simulator Investigation into Pitch Rate Command Systems in the Flared Landing Task

In 1983 an extensive TIFS investigation into pitch rate commands in the flared landing task was undertaken.[56,57] This study evaluated pitch-rate feedback with proportional and integral forward paths, rate command design, lead–lag pre-filters, superaugmentation, superaugmentation with lead–lag pre-filters, neutral static stability, and angle of attack and pitch-rate feedback required for level 1 conventional aircraft response. The aircraft configurations evaluated were a matrix constructed from seven aerodynamic models (three stable aircraft with different values of $1/\tau_{\theta_2}$, two neutrally stable aircraft with different values of $1/\tau_{\theta_2}$, a shuttle-like vehicle, and a shuttle-like vehicle with canards) and eight pitch axis FCSs (two proportional plus integrated pitch-rate feedback systems with different undamped short-

period frequencies ($\omega_{n_{sp}}$), superaugmented, conventionally augmented, three shuttle FCS variants, and one shuttle FCS variant with a time delay).

Results from this study included findings that current integral-proportional pitch-rate FCSs provided good attitude control, which is required for good performance in the flared landing task. In addition, the pilot needs cues to control flightpath precisely in the landing flare. These cues may come from pilot acceleration, stick deflections and forces, initial aircraft response, and longer term aircraft response. In addition, many techniques can be used to provide level 1 performance.

Interestingly, this study discovered that classical predictive criteria did not provide adequate prediction for the flared landing task, although a time-domain predictive criterion developed from this experiment did work well.

Total In-Flight Simulator Validation of the X-29 Control System

The TIFS was used in 1984 to examine the X-29 control system, with particular attention to power approach.[58] The X-29, with its forward-swept wing (Fig. 24), is a statically unstable fly-by-wire airplane with a digital primary FCS, a digital backup FCS, and an analog backup FCS. This vehicle has a canard and a strake flap in addition to the full-span flaperon and rudder. The canard, strake flap, and flaperon are used for pitch control; the flaperon alone for roll control. Ground simulation had raised questions about the flying qualities of the X-29 in power approach, with some indication that the lateral–directional gains and stick gearings might be unsatisfactory. A three-phase program was undertaken to examine these issues.
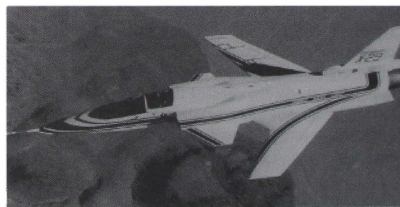
EC85 33297-001

Fig. 24 The forward-swept wing X-29 aircraft.

In the first phase, the originally proposed gains and stick gearing were examined in up-and-away and in power approach in the primary and both backup modes. Numerous PIOs led to reduction of the lateral–directional gains and the stick gearing in the primary mode and in the digital backup mode. The analog backup mode initially received only a limited evaluation because of a simulation anomaly, but the gains were modified and a corrected analog backup mode was evaluated. This corrected mode also demonstrated a

number of PIOs, but because of the limited data, no changes were made in this mode. The primary and digital backup modes were, however, modified with reduced gains and stick gearing.

Phase two of this study was a quick-look program that examined the design changes that resulted from phase one. Phase three, flown shortly before the first flight of the X-29, provided one last evaluation of the control laws in power approach, familiarization with the first-flight profile for the pilot and the control room personnel, and evaluation of selected emergency landing modes. The primary concern in this phase was the lateral PIO in the analog backup mode, which raised a safety-of-flight question. In this phase, flying qualities in all modes were found to be adequate for the first flight. The primary and digital backup modes, with their gains determined in previous testing, were found to exhibit level 1 and 2 handling qualities and the analog backup mode exhibited levels 1 to 3 handling qualities.

The X-29 aircraft was later flown at altitude in the analog backup mode to examine the lateral PIO seen in the TIFS study. Precision tasks, including bank angle captures and formation flight, were used to provoke any PIO. However, no lateral PIO tendencies were seen. This difference between the TIFS and the aircraft was attributed to errors in the predicted mathematical model of the X-29 and to the model-following techniques used to quicken the TIFS response, allowing this large airplane to fly like a fighter.

Total In-Flight Simulator Investigation of Proposed Shuttle Flight-Control System Modification

After the successful PIO suppressor study, another TIFS study was done in 1985 to examine further possible changes to the orbiter FCS.[54,55] In particular, a shaped pitch-rate feedback system, a command pre-filter and pure pitch-rate feedback equivalent system, and a C^* feedback system were compared to the baseline shuttle system. Additionally, reducing time delay in the FCS by moving the body bending filters from the command path to the feedback path was examined.

Although the addition of canards to the orbiter was not seriously contemplated, the use of canards was evaluated with the baseline and modified FCSs. Canards would have given sufficient control of the center of rotation so that the problems caused by pilot location would have been greatly reduced.

Learjet Flying Qualities Research

In the mid-1980's the Learjet was used for a limited in-flight simulation program that examined the effects of feel system dynamics on aircraft lateral handling qualities in the approach and landing task.[59] This study was sparked by the results of a brief study in the NT-33A aircraft.[59] In this Learjet study, two feel systems, one fast and one slow, were examined. The flight-control configurations had two possible transport time delays, designed so that the equivalent time

delay for the feel system and FCS combined were the same in each case. A baseline configuration with minimum overall time delay was also included. The tasks were bank angle captures and lateral-offset spot landings.

This study showed that the location of the time delay is important and that the feel system should be regarded as a separate dynamic element. Large overall time delay could be tolerated if a significant portion of the delay resided in the feel system. However, the same amount of overall time delay was unacceptable to the pilot if much of the delay was transport time delay downstream of the feel system. Additionally, this study indicated that the allowable time delay in the roll axis is a function of initial acceleration rate or "jerk."

The variable-stability Learjet has been used as a training tool at DFRF since 1983. Engineers are exposed to a training syllabus based on that used by the Air Force and Navy Test Pilot Schools.[11] All axes and modes are examined and stable, neutrally stable, and unstable configurations are flown. Time delays and feel system dynamics can also be varied. This aircraft has also been used by test pilots to review flying qualities areas.

NT-33A Investigation of Feel-System Characteristics on Roll Dynamics

In the late 1980's an investigation of the influence of lateral feel-system characteristics on fighter aircraft roll axis flying qualities was done with the NT-33A,[3] partly in response to the Learjet study of feel-system dynamics. This extensive study examined power approach, visual landing, and up-and-away tasks including formation, gun tracking, and computer-generated compensatory attitude tracking tasks displayed on the HUD. Experimental variations included the feel system frequency, force-deflection gradient, control system type (force or position input command), aircraft roll mode time constant, control system pre-filter frequency, and control system delay. The investigation was undertaken to determine how the feel system and the FCS interact and how the pilot assesses each.

The feel system is not equivalent to analogous control system elements in its influence on flying qualities. This led to the conclusion that flying qualities criteria should treat the feel system separately from the control system, since the feel system dynamics are apparent to the pilot and are not hidden in the total dynamics.

Investigation of Flightpath Control Using Throttles Only

In 1989 an airliner accident in which hydraulic power failed completely and differential thrust was used for flightpath control led to an investigation of the use of throttles for emergency flight control.[60] In addition to fixed- and moving-base ground simulations, this investigation included a cursory flight simulation program using the VSA Learjet, the F-15, and the PA-30 aircraft. In twenty minutes of flight using throttles only, the Learjet demonstrated

some control capability, with heading and altitude maintained within 500 ft. It showed good roll controllability with differential thrust and poor pitch control, with the phugoid being difficult to damp with throttle inputs.

The PA-30, a low-wing, twin-engine four-seat general aviation airplane, was difficult to control in all axes with thrust only. Gross control of the PA-30 was possible but landing on a runway would have been difficult.

The F-15, a twin-engine air superiority fighter, demonstrated good roll response and pitch response to throttle control. The F-15 rolled and banked well with throttle control only and a heading could also be held well. Altitude could be held within 100 ft at airspeeds below 200 kn, though phugoid damping was difficult.

These three experiments indicated that it is feasible to develop a control system for a large transport that would allow a safe return if hydraulic power were completely lost.

Concluding Remarks

There are two areas of major interest at the Dryden Flight Research Facility that simulation has addressed, which are landing fast, low-lift-to-drag ratio aircraft that cannot do go-rounds and pilot-induced oscillations in digital flight-control systems.

The first in-flight simulation program at the Dryden Flight Research Facility was an investigation of low-lift-to-drag ratio approach and landing characteristics, using the F-104 and the F-102A Delta Dagger aircraft. The most recent inflight simulation program here is an F-104 investigation into field of view requirements for the National AeroSpace Plane, a low-lift-to-drag ratio vehicle with limited visibility.

The X-15, the lifting bodies, X-20 Dynasoar, the shuttle, the National AeroSpace Plane–these low-lift-to-drag ratio airplanes land at high speeds and go-arounds are impossible. It has always been critical to get the landing pattern right before the flights. In-flight simulation has aided in the design of the pattern, the designation of high keys, approach angles, flare speeds, roundout altitudes, and touchdown speeds.

Structural and aerodynamic heating dictate small windows or remote viewing systems in hypersonic aircraft. The use of in-flight simulation answered questions on how small the windows could be, whether the remote viewing system needs to be stereoptic or monoptic, what resolution is required, what supplementary instrumentation is necessary and how to present this to the pilot, and a number of other display questions.

The interest of Dryden Flight Research Facility in aircraft with digital flight-control systems started around 1970 when the F-8 digital fly-by-wire program began. This aircraft, the first ever to be all-digital fly-by-wire, was used first as a demonstrator of the technology but it soon turned into a research tool examining digital flight-control system problems

like roll ratcheting. Interest in pilot-induced oscillations has always been high in the fast, high-performance research aircraft like the X-15 and the lifting bodies, as evidenced in part by the studies previously mentioned.

These two threads came together dramatically on October 26, 1977, when the Space Shuttle Enterprise, making a precision landing on the main runway at Edwards Air Force Base, experienced a fully-developed multiple-axis pilot-induced oscillation. As soon as the dust settled, Dryden Flight Research Facility began to use its experience in the investigation of flight-test problems.

The data were analyzed and the first Total In-Flight Simulator program confirmed that the causes were known. The effects of time delays in digital flight-control systems were examined in the F-8 aircraft. The pilot-induced oscillation suppressor filters were developed and tested in a second Total In-Flight Simulator program. While these filters were proven to work well, Dryden Flight Research Facility continued to examine improvements to the flight-control system for approach and landing and these proposed changes were examined in another Total In-Flight Simulator program.

Dryden Flight Research Facility then moved on from the practical, fix the problem and get the aircraft flying again approach, to research into more general issues, examining pilot-induced oscillation filters in fighter-type aircraft with the F-8 digital fly-by-wire and the NT-33A.

The location of the time delay, either in the feel system or in the flight-control system, was examined quickly in the NT-33A aircraft, more thoroughly in the Learjet, and exhaustively in a major NT-33A study. Thus a seemingly isolated incident led first to a solution to the incident and then to a body of research into the root problems.

References

[1] McNeill, Walter E., Gerdes, Ronald M., Innis, Robert C., and Ratcliff, Jack D., *A Flight Study of the Use of Direct-Lift-Control Flaps to Improve Station Keeping During In-Flight Refueling*, NASA TM X-2936, 1973.

[2] Swanborough, F.G., *United States Aircraft Since 1909*, Naval Institute Press, London, England, 1963.

[3] Bailey, R.E. and Knotts, L.H., *Interaction of Feel System and Flight Control System Dynamics on Lateral Flying Qualities*, NASA CR-179445, 1990.

[4] Berry, D.T. and Deets, D.A., "Design, Development, and Utilization of a General Purpose Airborne Simulator," AGARD Technical Report 529, May 1966.

[5] Breuhaus, Waldemar O., "The Variable Stability Airplane, From a Historical Perspective," *J. of American Aviation Historial Society*, Feb. 1990.

[6] Szalai, Kenneth J., Interview, Apr. 1991.

[7] Shafer, M.F., Smith, R.E., Stewart, J.F., and Bailey, R.E., "Flight Test Experience with Pilot-Induced Oscillation Suppression Filters," AIAA-83-2107, AIAA Atmospheric Flight Mechanics Conference, Gatlinburg, TN, Aug. 1983. See also NASA TM-86028, 1984.

[8] Edwards, John W. and Deets, Dwain A., *Development of a Remote Digital Augmentation System and Application to a Remotely Piloted Research Vehicle*, NASA TN D-7941, 1975.

[9] Duke, Eugene L., Brumbaugh, Randal W., and Disbrow, James D., "A Rapid Prototyping Facility for Flight Research in Advanced Systems Concepts," *IEEE Computer*, May 1989.

[10] Cohen, Dorothea and Le, Jeanette H., "The Role of the Remotely Augmented Vehicle (RAV) Laboratory in Flight Research," AIAA-91-2977, Aug. 1991.

[11] Ball, J., Berthe, C., Knotts, L., Parrag, M., and Smith, R., "Flight Syllabus and Background Material for the U.S. Air Force/U.S. Navy Test Pilot School Variable Stability Programs," Arvin/Calspan Advanced Technology Center, Buffalo, NY, Apr. 1981. Revised Jan. 1983.

[12] Matranga, Gene J. and Menard, Joseph A., *Approach and Landing Investigation at Lift-Drag Ratios of 2 to 4 Utilizing a Delta-Wing Interceptor Airplane*, NASA TM X-125, 1959.

[13] Matranga, Gene J. and Armstrong, Neil A., *Approach and Landing Investigation at Lift-Drag Ratios of 2 to 4 Utilizing a Straight-Wing Fighter Airplane*, NASA TM X-31, 1959.

[14] Swanborough, Gordon and Bowers, Peter M., *United States Navy Aircraft Since 1911*, Naval Institute Press, Annapolis, MD, 1976.

[15] Hallion, Richard P., *On the Frontier: Flight Research at Dryden, 1946-1981*, NASA SP-4303, 1984.

[16] Weil, Joseph and Matranga, Gene J., *Review of Techniques Applicable to the Recovery of Lifting Hypervelocity Vehicles*, NASA TM X-334, 1960.

[17] Armstrong, N.A. and Holleman, E.C., "A Review of In-Flight Simulation Pertinent to Piloted Space Vehicles," AGARD Technical Report 403, July 1962.

[18] Thompson, Milton O., "General Review of Piloting Problems Encountered During Simulation and Flights of the X-15," Society of Experimental Test Pilots (SETP) 9th Symposium Proceedings, Beverly Hills, CA, vol. 7, no. 4, Sept. 1965, pp. 70–77.

[19] Thompson, Milton O., Weil, Joseph, and Holleman, Euclid C., "An Assessment of Lifting Reentry Flight Control Requirements During Abort, Terminal Glide, and Approach and Landing Situations," Presented to AGARD Specialists' Meeting on Stability and Control, Cambridge, England, Sept. 1966. See also NASA TM X-59119.

[20] Dana, William H. and Gentry, J.R., *Pilot Impressions of Lifting Body Vehicles, Flight Test Results Pertaining to the Space Shuttlecraft*, NASA TM X-2101, 1970, pp. 73–88.

[21] Manke, John A., Retelle, John P., and Kempel, Robert W., "Assessment of Lifting Body Vehicle Handling Qualities," AIAA-71-310, Mar. 1971.

[22] Petersen, Forrest S., Rediess, Herman A., and Weil, Joseph, *Lateral–Directional Control Characteristics of the X-15 Airplane*, NASA TM X-726, 1961.

[23] Kidd, E.A., Bull, G., and Harper, R.P., Jr., "In-Flight Simulation—Theory and Application," AGARD Technical Report 368, Apr. 1961.

[24] Breuhaus, W.O., "Simulation of the X-15 Using the T-33 Variable Stability Airplane," Panel on Acceleration Stress of the Armed Forces, National Research Council Committee on Bio-Astronautics, Moffett Field, CA, Mar. 1961.

[25] F-104 Curatorial file, National Air and Space Museum, Smithsonian Institution, Washington, D.C.

[26] Matranga, Gene J., Dana, William H., and Armstrong, Neil A., *Flight-Simulated Off-the-Pad Escape and Landing Maneuvers for a Vertically Launched Hypersonic Glider*, NASA TM X-637, 1962.

[27] Perry, John J., Dana, William H., and Bacon, Donald C., Jr., *Flight Investigation of the Landing Task in a Jet Trainer with Restricted Fields of View*, NASA TN D-4018, 1967.

[28] Berry, D., Cox, T., and Shafer, M., "Landing Visibility: Flight Experience at NASA Ames-Dryden," Eighth National Aerospace Plane Technology Symposium Proceedings, Mar. 1990, pp. 12-1 to 12-15.

[29] Layton, Garrison P., Jr., and Dana, William H., *Flight Tests of a Wide-Angle, Indirect Optical Viewing System in a High-Performance Jet Aircraft*, NASA TN D-3690, 1966.

[30] Gaidsick, Harold G., Dana, William H., and McCracken, Robert C., *Evaluation of an Indirect Viewing System for Lifting-Body Terminal-Area Navigation and Landing Tasks*, NASA TN D-5299, 1969.

[31] Thompson, Milton O., Interview, Apr. 1991.

[32] Meeker, J.I., *Evaluation of Lateral–Directional Handling Qualities of Piloted Re-Entry Vehicles Utilizing Fixed-Base and In-Flight Evaluations*, NASA CR-778, 1967.

[33] Kempel, Robert W., *Analysis of a Coupled Roll-Spiral-Mode, Pilot-Induced Oscillation Experienced with the M2-F2 Lifting Body*, NASA TN D-6496, 1971.

[34] Matranga, Gene J., Mallick, Donald L., and Kluever, Emil E., "An Assessment of Ground and Flight Simulators for the Examination of Manned Lunar Landing," AIAA-67-238, Feb. 1967.

[35] Kluever, E.E., Mallick, Donald L., and Matranga, Gene J., "Flight Results With a Nonaerodynamic, Variable Stability, Flying Platform," Society of Experimental Test Pilots (SETP) 10th Symposium Proceedings, Beverly Hills, CA, vol. 8, no. 2, Sept. 1966, pp. 98–121.

[36] Szalai, Kenneth J., *Validation of a General Purpose Airborne Simulator For Simulation of Large Transport Aircraft Handling Qualities*, NASA TN D-6431, 1971.

[37] Szalai, Kenneth J., *Motion Cue and Simulation Fidelity Aspects of the Validation of a General Purpose Airborne Simulator*, NASA TN D-6432, 1971.

[38] Mallick, Donald L., and Fulton, Fitzhugh L., Jr., "Flight Crew Preparation and Training for the Operation of Large Supersonic Aircraft," FAUSST Technical Report 7.3.2.02, FAUSST VII Meeting, Mar. 1969.

[39] Hughes, Donald L., Powers, Bruce G., and Dana, William H., *Flight Evaluation of Some effects of the Present Air Traffic Control System on Operation of a Simulated Supersonic Transport*, NASA TN D-2219, 1964.

[40] Holleman, Euclid C., *Flight Investigation of the Roll Requirements for Transport Airplanes in Cruising Flight*, NASA TN D-5957, 1970.

[41] Holleman, Euclid, C. and Powers, Bruce G., *Flight Investigation of the Roll Requirements for Transport Airplanes in the Landing Approach*, NASA TN D-7062, 1972.

[42] Peterson, Bruce, Interview, May 1991.

[43] Powers, Bruce G., *Analytical Study of Ride Smoothing Benefits of Control System Configurations Optimized for Pilot Handling Qualities*, NASA TP-1148, 1978.

[44] Kock, Berwin M. and Fulton, Fitzhugh L., Jr., *Approach and Landing Studies, Flight Test Results Pertaining to the Space Shuttlecraft*, NASA TM X-2101, 1970, pp. 99–108.

[45] Hoag, Peter C. and Schofield, B. Lyle, IFR Experience with Unpowered, Low-Lift-Drag-Ratio Landing Approaches, Flight Test Results Pertaining to the Space Shuttlecraft, NASA TM X-2101, 1970, pp. 109–125.

[46] Holleman, Euclid C., *Summary of Flight Test to Determine the Spin and Controllability Characteristics of a Remotely Piloted, Large-Scale (3/8) Fighter Airplane Model*, NASA TN D-8052, 1976.

[47] Weingarten, Norman C., "In-Flight Simulation of the Space Shuttle Orbiter During Landing Approach and Touchdown in the Total In-Flight Simulator (TIFS)," Arvin/Calspan Advanced Technology Center, Buffalo, NY, Technical Report 6339-F-1, Sept. 1978.

[48] Berry, D.T., Powers, B.G., Szalai, K.J., and Wilson, R.J., "A Summary of an In-Flight Evaluation of Control System Pure Time Delays During Landing Using the F-8 DFBW Airplane," AIAA-80-1626, Aug. 1980.

[49] Berry, Donald T., Powers, Bruce G., Szalai, Kenneth J., and Wilson, R.J., "In-Flight Evaluation of Control System Pure Time Delays," *J. Aircraft*, vol. 19, no. 4, Apr. 1982, pp. 318–323.

[50] Weingarten, Norman C., "In-Flight Simulation of the Space Shuttle (STS-1) During Landing Approach with Pilot-Induced Oscillation Suppressor," Arvin/Calspan Advanced Technology Center, Buffalo, NY, Technical Report 6339-F-2, Dec. 1979.

[51] Powers, B.G., "An Adaptive Stick-Gain to Reduce Pilot-Induced Oscillation Tendencies," *J. Guidance and Control*, vol. 5(2), Mar.–Apr. 1982, pp. 138–142.

[52] Bailey, R.E. and Smith, R.E., *An In-Flight Investigation of Pilot-Induced Oscillation Suppression Filters During the Fighter Approach and Landing Task*, NASA CR-163116, 1982.

[53] Bailey, R.E., Smith, R.E., and Shafer, M.F., "An In-Flight Investigation of Pilot-Induced Oscillation Suppression Filters During the Fighter Approach and Landing Task," Society of Flight Test Engineers (SETP) Symposium Proceedings, New York, NY, Sept. 1982.

[54] Larson, Richard R., Smith, Rogers E., and Krambeer, Keith D., "Flight-Test Results Using Nonlinear Control With the F-8C Digital Fly-By-Wire Aircraft," AIAA Guidance and Control Conference Proceedings, Aug. 1983, pp. 97–110. See also AIAA-83-2174-CP.

[55] Weingarten, Norman C., "An In-Flight Investigation of Various Longitudinal Flight Control Systems in the Space Shuttle Orbiter During Approach and Landing," Arvin/Calspan, Buffalo, NY, Technical Report 7263-1, Dec. 1985.

[56] Powers, Bruce G. and Sarrafian, Shahan K., "Simulation Studies of Alternate Longitudinal Control Systems for the Space Shuttle Orbiter in the Landing Regime," AIAA Atmospheric Flight Mechanics Conference Proceedings, Aug. 1986, pp. 182–192. See also AIAA-86-2127.

[57] Berthe, C.J., Chalk, Charles R., and Sarrafian, S., *Pitch Rate Flight Control Systems in the Flared Landing Task and Design Criteria Development*, NASA CR-172491, 1984.

[58] Berry, Donald T. and Sarrafian, Shahan K., "Validation of a New Flying Quality Criterion for the Landing Task," AIAA Atmospheric Flight Mechanics Conference Proceedings, Aug. 1986, pp. 175–181. See also AIAA-86-2126 and NASA TM-88261, 1986.

[59] Sarrafian, Shahan K., and others, Memoranda and Internal Documents, 1985–6.

[60] Smith, Rogers E. and Sarrafian, Shahan K., "Effect of Time Delay on Flying Qualities: An Update," AIAA Guidance, Navigation, and Control Conference Proceedings, Aug. 1986, pp. 711–720.

[61] Burcham, F.W., Jr., Fullerton, Gordon, Gilyard, Glenn B., Wolf, Thomas D., and Stewart, James F., "A Preliminary Investigation of the Use of Throttles for Emergency Flight Control," AIAA-91-2222, June 1991.

THE ROLE OF THE REMOTELY AUGMENTED VEHICLE (RAV) LABORATORY IN FLIGHT RESEARCH

Dorothea Cohen*
and
Jeanette H. Le**
NASA Dryden Flight Research Facility
Edwards, California

Abstract

This paper presents an overview of the unique capabilities and historical significance of the Remotely Augmented Vehicle (RAV) Laboratory at the NASA Dryden Flight Research Facility. The report reviews the role of the RAV Laboratory in enhancing flight test programs and efficient testing of new aircraft control laws. The history of the RAV Laboratory is discussed with a sample of its application using the X-29 aircraft. The RAV Laboratory allows for closed- or open-loop augmentation of the research aircraft while in flight using ground-based, high performance real-time computers. Telemetry systems transfer sensor and control data between the ground and the aircraft. The RAV capability provides for enhanced computational power, improved flight data quality, and alternate methods for the testing of control system concepts. The Laboratory is easily reconfigured to reflect changes within a flight program and can be adapted to new flight programs.

Nomenclature

ASCII	American Standard Code for Information Interchange
CADRE	cooperative advanced research experiment
CL	control law
CRT	cathode-ray tube
FSW	forward-swept wing
HiMAT	highly maneuverable aircraft technology
JSC	Johnson Space Center
PCM	pulse code modulated
RAV	remotely augmented vehicle
RAVES	Remotely Augmented Vehicle Expert System
RCD	remotely computed display
RPV	remotely piloted vehicle
SRV	spin research vehicle
TACT	transonic aircraft technology
WATR	Western Aeronautical Test Range

Introduction

The NASA Dryden Flight Research Facility (Dryden) strives to enhance its flight test programs by providing superior data and testing new control law concepts safely and economically with high-quality results. Accomplishing these objectives efficiently calls for minimal flight hardware and software modifications and the use of computational power available from state-of-the-art ground-based computers.

In response to these needs, the Remotely Augmented Vehicle (RAV) Laboratory evolved to support flight test programs at Dryden. The RAV Laboratory's capabilities encompass three main areas: (1) research with a remotely piloted vehicle (RPV), (2) research with a RAV, and (3) remotely computed displays (RCDs) (Fig. 1). These remote computation techniques help reduce the time and cost necessary to complete a desired flight mission and improve the quality of flight research data.

Research with RPVs may be necessary due to flight safety reasons, cost limitations, or unavailability of a man-rated aircraft. The control laws for RPVs can be implemented in either the ground or onboard control law (CL) computers with a cockpit in the RAV Laboratory. Pilot inputs from the cockpit are fed through the ground-based CL computers and the resulting outputs are uplinked to the aircraft to drive the flight control surfaces. When the control laws are implemented onboard the RPV, the pilot's commands are uplinked to the aircraft. The visual cues for the pilot on the ground are provided through the RAV cockpit instruments and cameras either onboard the RPV or from a chase plane.

Remotely augmented vehicles provide new methods of testing control law system concepts by implementing the new control laws on the ground-based computers for more computational capability. Onboard primary control laws are then remotely augmented by ground computers which uplink surface commands to add to existing pilot inputs. The RAV computers can also uplink frequency sweeps, step inputs, or other commands for cleaner data collection than conventional methods. Individual surfaces can also be driven with this technique, which may not be possible solely through conventional pilot inputs to the onboard control laws.

The ground-based computers provide trajectory guidance through RCDs. These RCD techniques let ground computers use downlinked aircraft parameters to calculate the errors between the actual and desired flight conditions; the pilot may not be able to mentally compute such errors based on conventional flight instruments. These ground-computed directional cues are uplinked to a cockpit display to decrease the pilot workload required to achieve a desired flight condition. In some instances, RCDs make it possible to achieve flight profiles that could not have been attained by other means.[1]

History of the Remotely Augmented Vehicle Laboratory

The current RAV Laboratory capabilities have evolved beyond those of its predecessor, the Remotely Piloted Research Vehicle Facility created in 1971. In 1983, the Facility was combined with the Simulation Facility and incorporated remotely computed display techniques to form the Simulation/Remotely Controlled Vehicle and Display Laboratory. Presently, it is called the RAV Laboratory and it encompasses remote augmentation capabilities, whether open- or closed-loop, developed for a test vehicle.

The technology for remote control, augmentation, and displays were derived from programs like the F-15 spin research vehicle (SRV), the F-8 cooperative advanced research experiment (CADRE), and the F-111 transonic aircraft technology (TACT). The RPV, RAV, and RCD capabilities have since been applied, individually and collectively, to other flight test programs as well.

The need for an RPV capability emerged from the F-15 SRV program which investigated stall and spin characteristics. Due to high cost and the risks involved with a full-scale flight vehicle program, the project was conducted with a remotely controlled 3/8-scale prototype of the aircraft. Both the primary and secondary flight controls and pilot inputs were implemented through the RAV Laboratory.[2]

The F-8 CADRE was a remotely augmented vehicle used to develop nonlinear pitch flight control algorithms. The CADRE used FORTRAN control laws implementation by ground-based computers to avoid additions of onboard systems for the task. Also, since the control law algorithms were selected without previous pilot knowledge and initiated from the ground, the pilot evaluation of aircraft handling qualities was not biased.[3,4]

The F-111 TACT, designed with a supercritical wing for transonic maneuvers, was the first flight program to experiment with the remotely computed displays concept. Ground computers calculated the necessary trajectory correction based on downlinked flight data. This correction was uplinked to the aircraft to help the pilot accomplish the desired maneuvers. Better quality data were obtained with less time and cost.

Since these initial programs, the RAV Laboratory capabilities have been used in other flight test programs. The RPV techniques have been used in the highly maneuverable aircraft technology(HiMAT) program and the B-720 controlled impact demonstration, among others. Research vehicles like the X-29 forward-swept wing (FSW), the F-111 mission adaptive wing, and the F-18 high angle-of-attack research vehicle used RAV capabilities. These programs, along with others such as the F-15 highly integrated digital electronic control and the F-104 aircraft, also used RCD capabilities to help carry out their respective research.[1,5,6,7]

Remotely Augmented Vehicle Laboratory System in Flight Testing

The RAV Laboratory's role in flight testing is to support any flight test mission that requires closed-loop remote augmentation capabilities such as RAV and RPV applications, or open-loop capabilities as demonstrated in the use of RCDs (Fig. 2). The Laboratory provides the ground-based computational power necessary for remotely augmented vehicle missions. The RAV Laboratory is also equipped with its own raw data processing and flight monitoring capabilities.

The RAV Laboratory interfaces with other Dryden flight test facilities to accomplish its missions. The downlink signal is received from the Western Aeronautical Test Range (WATR), and the Laboratory sends back to the WATR the uplink parameters to be transmitted to the test aircraft (Fig. 3). The RAV Laboratory provides WATR with the processed downlink data and the calculated control law parameters for real-time recording. The Laboratory also interacts with the Dryden Mission Control Center, a facility within WATR which manages communication, conducts real-time analyses, and produces displays for flight safety.

Laboratory Hardware Description

The RAV Laboratory computer components include a data processing computer, a CL computer, and an expert flight monitoring system. Local recording hardware includes tape drives, printers, and strip charts (Fig. 3). A

MIL-STD-1553B bus control unit distributes data throughout the RAV Laboratory, while an uplink encoder and downlink driver permit access to WATR.

The RAV Laboratory uses two Encore 32/6750 (Encore, Fort Lauderdale, Florida) real-time computers with shared memory to handle the pulse code modulated (PCM) data processing and CL computations. These computers receive downlink telemetry from the aircraft through the WATR. The PCM computer converts raw parameter data into engineering units and exchanges the data with the CL computer through shared memory. The PCM computer also sends raw data to the monitoring terminals and the Remotely Augmented Vehicle Expert System (RAVES). These computers also have direct lines to the magnetic tape drive and a printer.

The MIL-STD-1553B bus is linked to the Encore shared memory region by a bus control/remote terminal unit and distributes the processed data to the expert monitoring system, strip charts, uplink encoder, and data formatter. Uplink data are sent from the uplink encoder through WATR transmitters to the test aircraft. Downlink data from the PCM computer and CL-computed parameters are sent through the data formatter to WATR to be recorded. Communication with both the pilot and other flight personnel is possible through a radio network handled through Mission Control.

The RAVES is interfaced to the 1553 bus by way of a decommutation system. The decommutation system monitors the raw PCM data stream and the 1553 bus to provide RAVES with the necessary flight and synchronization data for telemetry monitoring. The decommutation system also provides limited mathematical capability for the preprocessing of data for RAVES. The workstation can command the decommutation system using a serial port link.

Pulse Code Modulated and Control Law Software

The PCM software is designed to decommutate and calibrate downlink and uplink flight data, while the CL software calculates the control law algorithms or supplementary guidance systems to implement on the research aircraft. Each software is built from a generic skeleton that can be specifically reconfigured to operate with a specific flight program. The code structure includes a background task that does initialization and dynamically updates the cathode-ray tube (CRT) display pages on a time-available basis. These display pages output parameter information to give the operator a way of interfacing with the software systems. Each software also manages an interrupt-driven real-time loop which carries out the higher priority real-time tasks.[8]

The PCM real-time tasks include synchronization checks, parameter decommutation and calibration, downlink discrete processing, and outputting of uplink parameters (Fig. 4). The CL real-time loop inputs the downlink parameters and executes the necessary front-end calculations,

validity tests, and mission-specific tasks such as guidance needle calculations or control surface inputs (Fig. 5).

For flight programs that require RAV capabilities, the desired downlink or uplink parameters are pre-specified. The PCM computer extracts these parameters from the frames of data telemetered between the test aircraft and the ground.

Information necessary to decommutate and calibrate these specified parameters are contained in an input file which is read by the PCM software during initialization. This file is generated for each flight program and is updated and re-released to accommodate changes. The instructions for each parameter include the parameter name, sampling rate, frame of data where it first appears, word position within the frames, and buffer destination. For the calibration, the PCM software also requires the type of parameter (uplink or downlink), method of calibration to be conducted (curve fit or tabular), and corresponding calibration scaling.

Discretes require no calibration and are loaded into a discrete buffer to be processed and sent to the CL computer as an integer array. The PCM computer also receives radar data from the WATR which are converted to integer format in a separate interrupt-driven loop.

The CL software ground-computes the alternate control law algorithms implemented for the test aircraft. For RAV research, the CL computer can calculate pre-determined surface commands such as frequency sweeps, pulses, or individual surface deflections to add onto the existing pilot inputs. For RPVs, the CL computer handles the control laws for the aircraft as it is piloted from the ground and uplinks the output commands to drive the corresponding onboard control surfaces. For RCDs, the CL software determines differences between desired flight conditions and the actual conditions from the downlinked flight data. These differences are then uplinked and displayed onboard the test aircraft.

The PCM and CL computers perform real-time tests to track the data transfer status between the aircraft, the RAV Laboratory, and other systems in the loop. Each computer observes the execution status of the other system, performs data synchronization checks, and monitors data transfer failures between the two computers.

The PCM computer passes its interrupt counter through shared memory to the CL computer to indicate whether or not its real-time loop is still executing. Should the PCM fail, flags will be set in shared memory and subsequently sent to the uplink encoder to inform the aircraft of its downlink and uplink status with the RAV Laboratory. Likewise, the PCM computer can monitor the CL real-time loop counter which is placed in shared memory.

Synchronization checks determine if the PCM computer is in sync with the PCM data stream. If the sync fails, the

downlink data stream processing is bypassed and a flag signals the aircraft and the CL computer of the situation.

The PCM computer also performs window tests and limit checks on the incoming data to ensure that the data do not fall outside the specified range.

Other tests that are conducted by the CL software include wraparound tests, pilot override, radar sync, and any maneuver/mode/limits/reasonableness tests that vary with the flight mission. These validity tests are conducted before the calculated outputs from the CL computer are placed in shared memory.

All RAV software is validated to confirm that the software meets its specification, and verifications are conducted to show that the software running is the correct version. Software validation for the PCM software can be done with the use of pre-recorded flight data (in the case of an existing flight program), actual flight data, or other ways of generating an input PCM stream such as a PCM formatter to provide "artificial" downlink data. The downlink data are fed into the PCM software that carries through the conversions. Data output from the PCM software are then monitored from the CRT display pages. The CL software is validated by interfacing with a real-time simulation of the aircraft. Combined systems tests are conducted, incorporating all of the hardware into the loop with the aircraft on the ground. Once validated and released for use, the software is then verified by check sums produced by a cataloger and bit comparisons to the same program on a tape.

A section of pre-flight checks is also dedicated to verification of RAV operations. These checks ensure that the aircraft link to RAV is operational, that the RAV laboratory is sending and receiving commands, and that the pilot can disengage RAV if necessary.

Remotely Augmented Vehicle Expert System Software

The RAVES in the RAV Laboratory was designed to make the monitoring of RAV flights easier. The system gives the operator more effective displays using color thresholds, graphical representations of aircraft systems, and other visual cues. Before RAVES, the displays used for flight monitoring in the RAV Laboratory were limited to black and white CRT displays with ASCII outputs and no mouse control. Newer technology has upgraded the monitoring capabilities of criticalparameters in the Laboratory by providing color graphics and mouse capabilities to improve human-computer interface (Fig. 6).[9]

The RAVES is based upon the Johnson Space Center's (JSC's) Real-Time Display System, a C-language program that supports the space shuttle.[10] Benefits arise from using the JSC software because (1) concurrent developments by

NASA, JSC, and the Air Force can be shared to improve the system, (2) a large base of users is already associated with the software, and (3) the collective effort fosters collaboration among the government branches. The programming methodology of the data acquisition, a round robin-style ring, is taken from JSC's program, along with some of the graphics for monitoring the data acquisition. The actual data acquisition is through a decommutation system with the ability to monitor the raw PCM and 1553B bus.

The RAVES display graphics for flight parameter monitoring uses DataViews[®], a generic graphics package. DataViews[®] includes two packages: DVDraw[®], an interactive drawing package that allows for dynamics; and DVTools[®], the programming graphics language. DVDraw[®] lets users design new displays without the demand for coding by a programmer. Modifications of an existing display or creation of a new display can then be quickly done without recompiling the RAVES software.

The RAVES has four major display features: a monitoring window, a status line, a fault message window, and an expert system interface. The primary window displays user-designed pages that monitor flight parameters. The status line shows the current flight number, date, time, and data acquisition status. The fault message window has time-tagged and color-coded messages that are also logged to a file. The expert system includes notification of a dangerous or unusual occurrence within the fault message. Also, as an optional feature, the system lists appropriate actions to take in a popup window.

The RAVES uses a standardized color-coding technique to indicate status for critical and noncritical parameters as well as selection modes for various features (Fig. 7). Colors let the monitoring engineer quickly note flight status. This standardization also lets engineers go from project to project with the same color scheme, thereby lowering the learning curve on a new project.

The RAVES also has a local data-logging feature. This feature, if enabled, allows RAVES to continuously take data received and write it to a round robin file. When an interesting condition occurs, the user can tell RAVES to save the current contents of the round robin file to a permanent log file of selectable size. In addition to this log file, RAVES also creates three files containing the time of the save, parameters accessed by direct memory, and corresponding fault messages. Up to 10 unique sets of log files can be saved for each run of RAVES.

Tests are conducted for RAVES to verify the calculation algorithm for each parameter, the fault messages, the parameter limits, all changes of color, the changes of sign, and the expert system messages. This series of tests follows on the completion of PCM and CL verification and validation.

Laboratory Application in Flight Testing: The X-29 Flight Program

The X-29 FSW program, a study of new flight control laws and aerodynamic concepts, relied on the RAV Laboratory's abilities to remotely augment vehicle configurations and uplink remotely computed displays. The RAV Laboratory's computers provided a series of guidance parameters for an onboard RAV pilot steering system to help the pilot achieve desired flight maneuvers and control. The RAV systems also added pre-calculated control surface and stick or pedal commands to the existing pilot inputs for studies in aircraft flight responses.

When assisting the pilot in the guidance task, the CL computer compared the actual flight data to a reference flight profile as selected by the ground operator. The CL computer then calculated the differences and uplinked them to a set of vertical, horizontal, and Mach guidance needles onboard the aircraft to cue the pilot.

The CL computer also generated a series of pre-set input commands for uplink to the aircraft control surfaces, pitch or roll stick and directional pedal. The X-29 program must be able to run a series of different step inputs sequentially to the same or different surface or pilot command. These step inputs can be selected by the ground operator from a CRT menu of pre-calculated and pre-tested step inputs once RAV has been enabled by the pilot. Similarly, frequency sweeps can either be initiated by the ground operator or engaged by the pilot. Such ground-computed inputs are added to the existing pilot commands and therefore the pilot is never removed from the loop. These remote inputs also give cleaner flight data results.

The RAVES was first used during the X-29 flight test program to monitor the RAV operations. Engineers were able to detect RAV operation failures more quickly with the program's graphics capabilities. The RAVES also offered a practical local recording technique in its logging feature.

The most pronounced benefit of RAVES is in the speed at which the engineer can see, and therefore react to, a failure. Before RAVES, a failure could be noted in two ways: by a light-emitting diode light on a control panel, or by a logical discrete (true or false) on an ASCII display. The RAVES offers two alternate methods to quickly recognize failures: a color change on the display, or a notice posted in the fault message window. For example, RAVES signaled a RAV command uplink failure by changing the color of the parameter from green to redand outputting a red fault message. In another instance, the incorrect radar switch feed was discovered through fault messages.

The RAVES logging feature allows quick access to critical flight segments to help determine what may have occurred during the flight. Also, logging allows data files to be saved and replayed without the need of a flight tape and having the entire laboratory operational. In one instance, a flight was aborted because of failures to maintain RAV communications with the X-29 test aircraft. A log made during flight was replayed through RAVES, requiring only the presence of the RAVES workstation where the data file resides. The cause for failure was easily detected and flight resumed the following day.

The RAV capabilities provided the X-29 program with a way to accomplish missions which may not have been conceivable otherwise. Manipulation of individual surfaces, for instance, would not have been possible solely through pilot inputs because the primary control laws move multiple surfaces simultaneously. Cleaner aircraft responses were also obtained through the ground-computed inputs. Also, the steering guidance cues gave pilots an easier and faster way of accomplishing the desired flight profiles. The RAV systems provide for efficient flight test operations and high-quality research data at the lowest possible costs.

Systems Development, Adaptation, and Integration for New Flight Programs

Adapting RAV operations to a new flight test program requires both software and hardware modifications to reflect the aircraft and its mission. Because the foundation for the RAV software packages are already established, any changes need only reflect those required by the flight test program. Hardware modifications are necessary for the aircraft to accommodate basic RAV operations, with more extensive ground hardware modifications necessary for RPV purposes.

Modifications of data processing information necessary for the PCM software are required for the flight program's specific needs. The PCM software engineer regenerates the data file that contains the instructions for decommutation and calibration of each flight parameter to include instructions for the new research program. Validation tests are again conducted to assure "flight-readiness" of the software for actual flight operations.

The CL software modifications may be more extensive and are highly dependent on the flight test objectives. Although the skeleton of the CL software remains the same, new front-end calculations are implemented to accommodate RPV, RAV, or RCD applications. The CL software can emulate the control system, be a separate system with augmentation, or take the downlinked onboard outputs, process them, and then send them back to the plane (bypassing the onboard system). Validation for this software is primarily done through the simulation and is further checked along with the PCM software during the combined systems tests.

There are three major efforts needed to add a new flight program module to RAVES: reprogramming the decommutation system, designing new displays, and calculating any parameters specific to the flight pro-

gram. Once these are completed, it is only necessary to test and verify the accuracy of the displays and parameters.

The RAVES displays can be designed by a project engineer on any workstation running DVDraw®. The decommutation system requires programming of the parameter list, which can be saved to diskette. A programmer takes the designed displays and parameter release documents for the project to devise the main menu display and code parameter calculation with appropriate messages. Once the updates are completed, the programmer and other engineers on the project will decide if there are any unusual conditions or sets of conditions to beware of. This knowledge is then integrated into the expert interface and fault message window.

Onboard software modifications are program dependent. The software hooks for RCD, RAV, and RPV applications must be added to remotely drive the vehicle instruments and surfaces. In RCD, the software is modified to take the uplink signal and display it on an instrument as a "fly-to" indicator. For RAV and research with RPV, the onboard CL software is adjusted to take the uplink and conduct scalings, limit checks, and error detections before applying the signal to the surface. There also is a provision in a piloted RAV for an automatic and pilot disengage of RAV Laboratory commands.

Remotely piloted vehicle operation will also need additional ground hardware to provide redundancy as well as a ground station for remote piloting. The redundant system consists of an additional set of PCM/CL stations and computers that serve as the standby system if the active computers fail. The pilot station usually consists of a cockpit with full instrumentation and video equipment for additional visual support. The stick computer allows selection of the stick and rudder characteristics and provides the interface between the pilot's controls and the CL computers.

Hardware modifications onboard the test aircraft are necessary for all RCD, RPV, and RAV applications. For any application by the RAV Laboratory, a dedicated receiver is installed on the aircraft to acquire the uplink signal. Sometimes a frequency (diversity) combiner is used to maintain blanket coverage for continuous uplink contact with the aircraft.

Modification for use of RCD involves using the uplink to drive displays onboard the aircraft. The output of the uplink on the aircraft is tied to a display device on the aircraft. This device may be a pre-existing onboard instrument or the aircraft may require the installation of a new display mechanism.

For RAV missions, the output of the uplink is fed into either an autopilot or the control system. Hardware provisions are needed to receive and feed these commands into the con-

trol system. Hardware and software systems checks ensure control and flight safety.

Aircraft modifications for RPV operation are similar to those in RAV operation. In RPV applications, the combiner is always used and there are more systems checks in both the hardware and software to ensure control and safety. The RPV operation has a ground-based backup system in case of a failure in the primary system. Visual cues are provided by cockpit instruments and the camera onboard the RPV or the chase plane. A backup system can also be placed onboard the RPV in case the uplink signal from the ground is lost. This onboard backup can be controlled from a chase aircraft which allows the chase pilot to land the RPV or direct it away from any populated area. If both systems are employed, the control authority can be switched between the ground or the chase plane backup system, as required.

Future Applications and Expansion of Capabilities for RAV

Other applications and expansions of RAV capabilities are continually being explored. Currently, computer graphics is being examined.

Aside from computer visuals, which only indicate aircraft trajectories on a map, new 3-D computer graphics showing an aircraft model and its flight attitudes have been designed and interfaced with ground simulations (Fig. 8). These visualization capabilities are easily transferable to the RAV Laboratory. The RAV data can be routed to the 3-D visual system to show the aircraft's motions during flight.

The future for RAVES includes the making of displays on-the-fly and an X-window interface. These capabilities are available in another package using the same graphics and operating in conjunction with the current Dryden simulations.

Also, two-way communication and control, currently supplied by the CL displays, can also be developed for the RAVES workstation by way of the decommutation system. This capability would allow users to command as well as monitor flights from RAVES.

Concluding Remarks

The Remotely Augmented Vehicle Laboratory at the NASA Dryden Flight Research Facility is unique in its ability to offer remotely piloted vehicle, remotely augmented vehicle, and remotely computed display capabilities in a single facility. These capabilities have helped the Facility to efficiently conduct its flight test programs, to provide better quality data, and to conduct tests that may not have been possible otherwise.

The ground-based computers in the Remotely Augmented Vehicle Laboratory add computational power and minimize the aircraft software and hardware modifications

required for flight testing of a research vehicle. Remotely piloted vehicle technology has allowed flight testing on vehicles such as the highly maneuverable aircraft technology vehicle where advanced aircraft systems required unmanned operations. Remote augmentation of vehicles offers a rapid method to test alternative control law concepts. Remotely computed displays have proven invaluable by assisting the pilot in performing difficult flight test maneuvers.

The wealth of past flight test experience with the Remotely Augmented Vehicle Laboratory has demonstrated the Laboratory's ability to quickly accommodate new research programs with a minimum of time and effort.

References

[1] Meyer, Robert R., Jr., and Schneider, Edward T., *Real-Time Pilot Guidance System for Improved Flight Maneuvers*, NASA TM-84922, 1984.

[2] Petersen, Kevin L., "Review of the 3/8th Scale F-15 Remotely Piloted Research Vehicle Program," presented at Meeting No. 38 of SAE Aerospace Control and Guidance Systems Committee, Sept. 22–24, 1976. Also published as NASA TN D-8216, 1976.

[3] Larson, Richard R., Smith, Rogers E., and Krambeer, Keith D., "Flight-Test Results Using Nonlinear Control With the F-8C Digital Fly-By-Wire Aircraft," AIAA 83-2174, Aug. 1983.

[4] Shafer, Mary, *Flight Investigation of Various Control Inputs Intended for Parameter Estimation*, NASA TM-85901, 1984.

[5] Petersen, Kevin L., "Flight Control Systems Development of Highly Maneuverable Aircraft Technology (HiMAT) Vehicle," AIAA 79-1789, Aug. 1979.

[6] Horton, Timothy W., and Kempel, Robert W., "Flight Test Experience and Controlled Impact of a Remotely Piloted Jet Transport Aircraft," NASA TM-4084, 1988.

[7] Kempel, Robert W., Phillips, Paul W., Fullerton, C. Gordon, and Bresina, John J., "Operational Flight Evaluation Technique Using Uplink Pilot Command Cues," presented at the 20th Annual SFTE Symposium, Reno, NV, Sept. 18–21, 1989.

[8] Evans, Martha, and Schilling, Lawrence J., *The Role of Simulation in the Development and Flight Test of the HiMAT Vehicle*, NASA TM-84912, 1984.

[9] Mackall, Dale, McBride, David, and Cohen, Dorothea, *Overview of the NASA Ames-Dryden Integrated Test Facility*, NASA TM-101720, 1990.

[10] Muratore, John F., "Trends in Space Shuttle Telemetry Applications," *Proceedings of the 1987 International Telemetering Conference*, San Diego, CA, Oct. 1987, pp. 283–290.
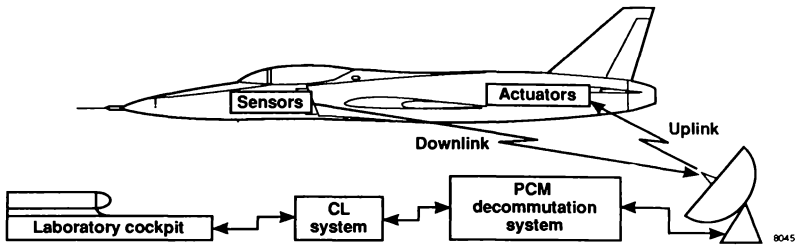
(a) Remotely piloted vehicle.
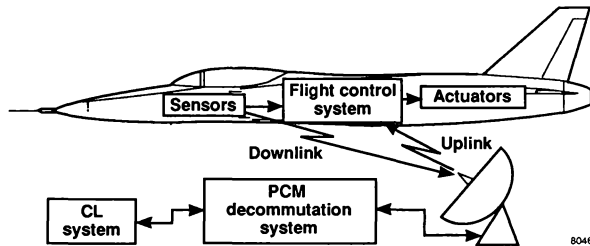
(b) Remotely augmented vehicle.
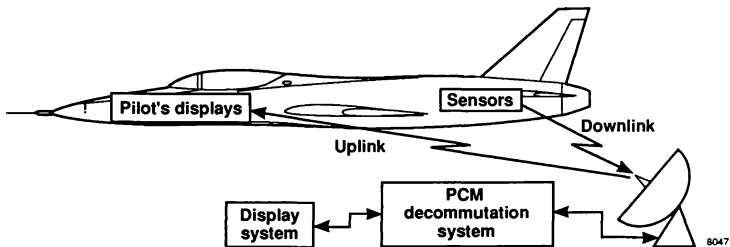
(c) Remotely computed display.

Fig. 1. RAV configurations.

Downlink

Uplink

Telemetry system

Cockpit

**Ground-based
flight computers**
- **Control laws**
- **Monitoring**
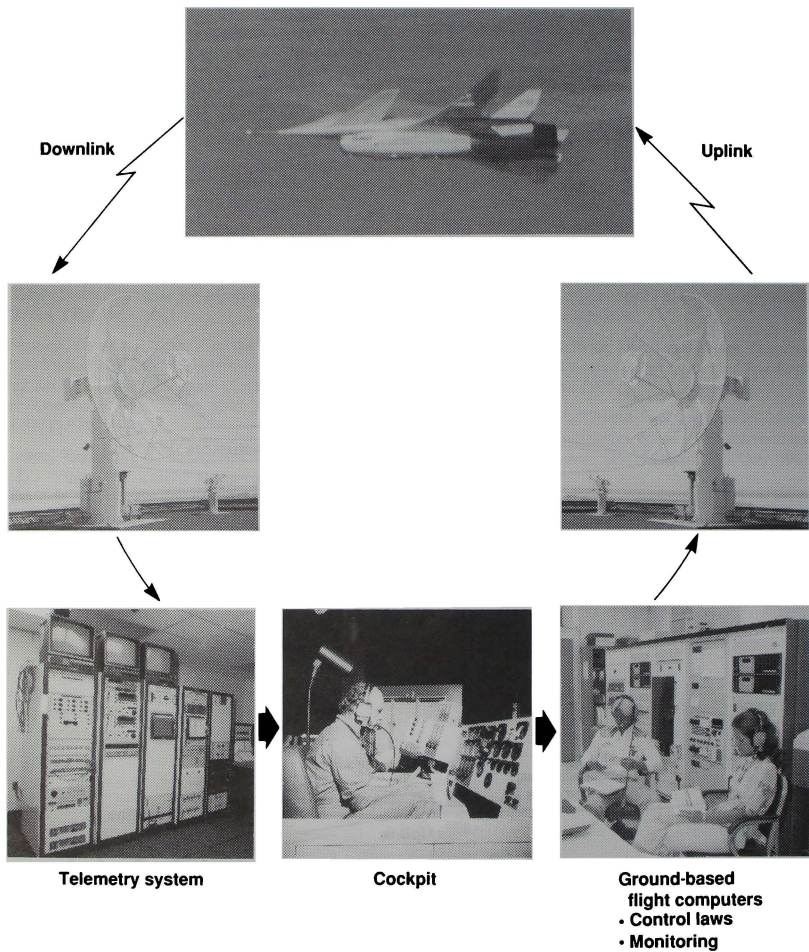
Fig. 2. RAV laboratory elements.
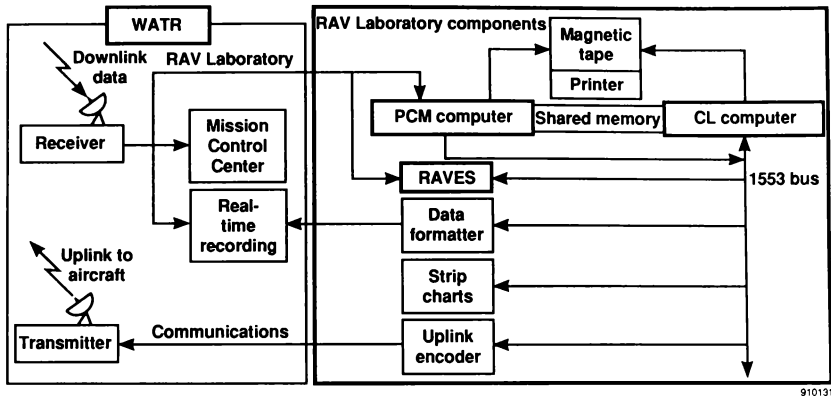
Fig. 3. RAV Laboratory block diagram.
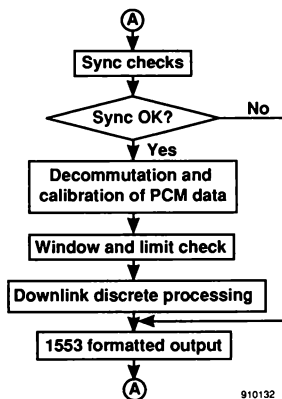
Fig. 4. PCM software structure.
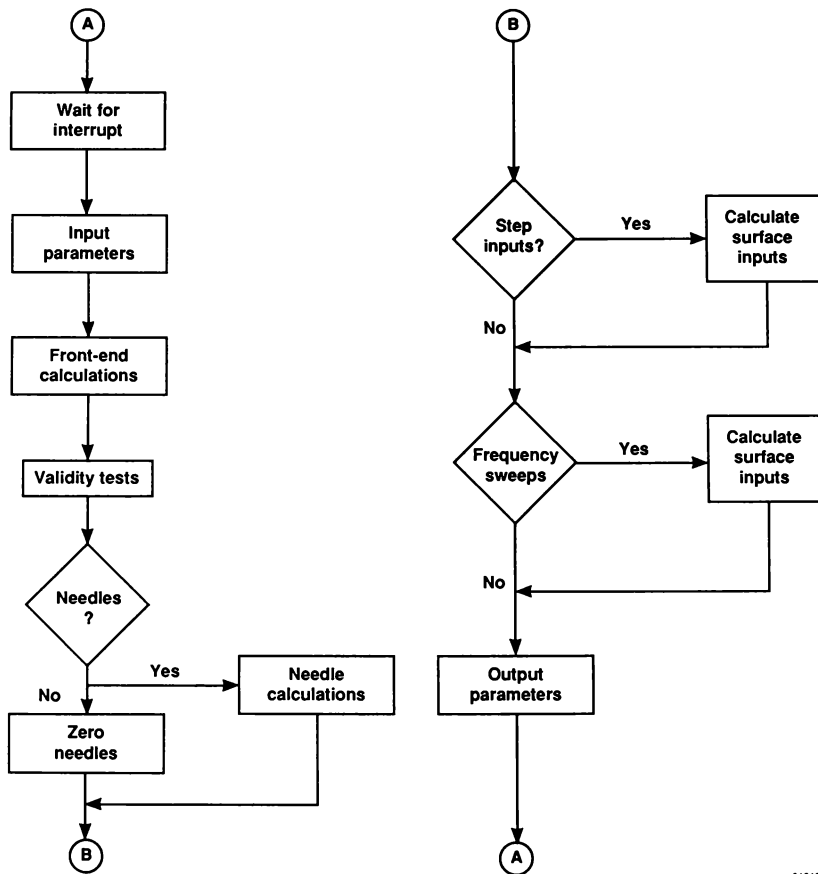
Fig. 5. CL software structure for full RAV application.

910133

ALL DISPLAY

```
FSW RAV FLIGHT RAVVAL: T  5 ZER0 ALL GAINS        REV: RAV00004  15:08:41

      INTRCL          0 CALCULATED VALUES        F SRFLIM F   LNED    T
   F  INTRPC          0  F HPT      ,0000E+00     F REAS0N F   LSTEP   F
   F  INTRRD          0     XMACHT  ,0000E+00     F HRD0VR F   LFREG   F
   F  ISYNCR          0     GBAR    ,0000E+00     F NZLIM  F   LRUNUP  F
   F  IDSC2D    0000        ALPHAT  ,0000E+00     F NZFAIL F
      IDSC2U    0000        KNTDLY          0     F DISSSM F
   D0WNLINK DATA                                  F ANASSM F
  1 F PAD     ,0000E+00                             SYNCF  F
  2 F GCD     ,0000E+00  RADAR DATA              F ARM0CE F
  3 F ANZD    ,0000E+00  1 F RADT   ,0000E+00    F DRM0CE F
  4 F PD      ,0000E+00  2 F RADX   ,0000E+00    F DEGNRM F
  5 F ALPD    ,0000E+00  3 F RADY   ,0000E+00    F RATLIM F
  6 F THAD    ,0000E+00  4 F RADZ   ,0000E+00      I0FAIL F
  7 F PHID    ,0000E+00  5 F RADVX  ,0000E+00    F XFER0K F
  8 F BTAD    ,0000E+00  6 F RADVY  ,0000E+00    F XFERFL F
  9   ADIMD   ,0000E+00  7 F RADVZ  ,0000E+00      LRV0VR F
 10   RVSURD  ,0000E+00  OPLINK DATA               RAVACT F
 11   ADIHD   ,0000E+00  1 F XMACH  ,0000E+00      RAVVAL T
 12   ADIVD   ,0000E+00  2 F SURF   ,3688E-39      W0WALL F
 13   AGC1    ,0000E+00  3 F H0RIZ  ,0000E+00      PID0FF F
 14   AGC2    ,0000E+00  4 F VERT   ,0000E+00      RAVENG F
```

(a) Control law display in standard black and white ASCII format.
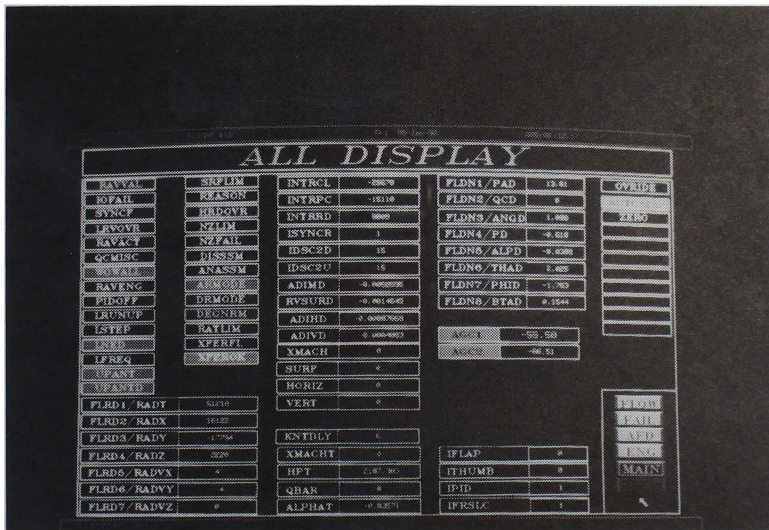
EC90 0010-006

(b) RAVES display.

Fig. 6. RAVES display in contrast to a CL display page.

35

Parameter type / Status	Critical	Noncritical
OK	GREEN	BLACK
Failing	YELLOW	YELLOW
Failed	RED	RED

Mode	Selected	Not selected
Color	BLUE	BLACK

(a) Designated colors highlighting critical and noncritical flight parameters based on status.

(b) Color scheme used for flight modes showing selection status.

Fig. 7. RAVES color-coding scheme.
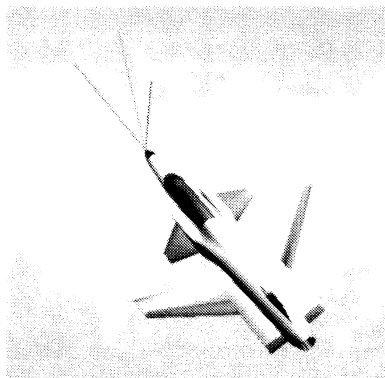
EC90 0084-2

Fig. 8. Computer-generated 3-D model of the X-29A aircraft showing its flight attitudes.

VISTA: AN ESSENTIAL TOOL FOR THE FUTURE

David E. Frearson
USAF Wright Laboratory
Wright-Patterson AFB, Ohio
and
Philip A. Reynolds
Calspan Corporation
Buffalo, New York

ABSTRACT

The Variable Stability In-Flight Simulator Test Aircraft (VISTA) is being developed by the United States Air Force (USAF) to replace an NT-33A aircraft, which has been widely used as an in-flight simulator since 1957. Development is being accomplished through a major contract awarded to the General Dynamics Corporation with Calspan Corporation as a major subcontractor. VISTA, based on a new F-16D aircraft, will uniquely support future aircraft development, test pilot training, flight control/display research, and avionic integration research. The VISTA configuration provides the realism, hardware and software flexibility, experimental control, and growth potential essential to long term, cost effective utility. The F-16 offers a substantial improvement in performance over existing in-flight simulators. Increases in speed, altitude, normal acceleration, and thrust-to-weight provide a significantly larger simulation envelope, enabling duplication of a wider range of fighter trajectories. VISTA will provide a high performance, affordable solution to high confidence, low risk aircraft systems testing. USAF's Wright Laboratory will own and manage VISTA through an independent contract operator. It is expected to be available to interested government and industry customers on a cost reimbursement basis before the end of 1992.

BACKGROUND

In-flight simulators have had a long and rich history of support to aerospace research and development. The NT-33A has simulated nearly every new US high performance aircraft introduced since 1957 and was the primary tool used in the development of today's flying qualities specifications.

The NT-33A utilizes a hybrid response feedback variable stability system (VSS) independently controlling three degrees of freedom. It is configured with the evaluation pilot in the front seat and the safety pilot in the rear seat. It has variable feel centerstick and sidestick controllers, and a programmable head-up display (HUD) in the forward cockpit. Performance-wise, the NT-33A is capable of about 400 knots, 35,000 feet altitude, and 4.5 g load factor.

An NT-33A program is currently under Air Force sponsorship to determine if proposed HUD symbology is safe and usable as the primary flight reference for unusual attitude recognition and recovery, instrument landing system approaches, and instrument flight tasks in fighter aircraft. This work is part of the HUD standardization program and involves comparative evaluation against electro-mechanical head-down instruments for identical tasks.

When one of the Saab-Scania JAS-39 Gripen aircraft was lost early in the test program, Saab undertook an extensive review of the flight control system design. The NT-33A was flown in 1989 to evaluate the original design and help develop improvements. Over 350 landings were made by 13 pilots. The final result was a Level 1 primary digital flight control system and a high degree of confidence in the result. (Reference 1).

The Lockheed/General Dynamics/Boeing Advanced Tactical Fighter design team sponsored an in-flight simulation program for the YF-22A prior to first flight. The aircraft was simulated in the power approach configuration with several aerodynamic and center-of-gravity conditions. The simulation included the digital flight control system with all gain schedules and gear up/down logic. A simplified

version of the YF-22A HUD provided the pilot with the flight information of the simulated aircraft. Twelve evaluation flights were flown.

In 1976, the Air Force and Navy Test Pilot Schools each incorporated stability and control instructional flights, using the NT-33A, into their curricula. These flights help new test pilots, who are already experienced military pilots, to understand stability and control variations during large maneuvers and high-g load factors. To date, over 400 Air Force and Navy test pilots have been trained in the NT-33A.

The USAF's other in-flight simulator, the Total In-Flight Simulator (TIFS) has been operational since 1971. TIFS, a modified C-131H, utilizes a digital model-following VSS to independently control the six degrees-of-freedom. Modifications made to the airframe and control systems to support the six degrees-of-freedom control include the addition of side force surfaces, direct lift flaps, and thrust control. TIFS features a side-by-side evaluation cockpit, attached to the NC-131H nose, with variable feel controls and simulator controlled displays.

Real insight was gained in landing simulations of the YF-23 using the NC-131H. Also, the simulation of the refueling condition showed an excellent pilot-aircraft combination which was almost too good to be believed until it was substantiated by flight tests in the actual aircraft. The importance of positive speed stability was proven in power approach. The flight control system was thoroughly stressed in landing, adding to pilot confidence.

In 1984, the NC-131H was used to obtain actual flight data on a reconfigurable flight control system. A fault detection algorithm was tested and the control laws reconfigured to roll with differential flap rather than aileron. The software ran in real-time on the standard in-flight simulation computers.

The need to replace the NT-33A arises from the fact that it is no longer representative of current and future high performance aircraft, and is increasingly difficult to logistically support. Its speed, altitude, and load factor limitations are increasingly confining it to simulations in the power approach regime. As the oldest flying tail number in the USAF inventory (and the only flying T-33), replacement parts have also become an issue. Parts are not available through normal supply channels and must be obtained from independent vendors. Although it has carved an enviable niche in aviation history, it is clear that the end of the NT-33A's useful life as a high performance in-flight simulator is at hand.

SYSTEM DESCRIPTION

This section updates the descriptions given in References 2 and 3 which were written at the beginning of VISTA development in 1988 and after preliminary design in 1989.

There are two major activities involved in the development of the VISTA: production of the selected host aircraft and its modification into an in-flight simulator. Production of the selected Block 30 F-16D is based on the Israeli Air Force Peace Marble II configuration. It includes heavy weight structure and landing gear and a large dorsal fairing to house avionics. Unnecessary offensive and defensive avionic and weapon systems have been deleted. The aircraft is equipped with the F110-GE-100 engine and an APG-68 radar. A Block 40 digital flight control system has been substituted for the normal Block 30 analog system, necessitating substitution of some Block 40 avionics as well. Even in a production sense, VISTA is one-of-a-kind.

Extensive modifications have been made to the production F-16D to convert it into an in-flight simulator and provide for its growth during its anticipated 25 year service life. The major areas of change involve the cockpit, the hydraulic system, integration of the response feedback variable stability system, and addition of a data acquisition system. The VISTA general arrangement is shown in Figure 1.

Both forward and aft cockpits have been extensively modified. The safety pilot will occupy the aft cockpit of VISTA. Besides commanding the flight, he will control the VSS configuration and evaluation pilot experimental setup. This necessitated relocation of primary controls and displays to the rear cockpit from their normal position in the front cockpit, and the addition of controls and displays necessary to operate and manage the VISTA simulation system. The aft cockpit layout is depicted

in Figure 2. The forward, or evaluation cockpit is equipped with the standard F-16 sidestick, rudder pedals, and throttle assembly. A number of control and display changes have been made to interface with the simulation system. The most notable new feature is a modular variable feel centerstick which may be quickly installed or taken out depending upon the experiment. Forward cockpit layout is shown in Figure 3.

Hydraulic system modifications were primarily driven by growth considerations. The prospect of additional force generating surfaces and the projected rate and capacity requirements resulted in a distribution system design that would handle flow rates of up to 100 gallons per minute. That meant larger tubing and higher capacity pumps. The decision was made to incorporate these changes during initial fabrication given the improbability of subsequent retrofit. The F-16's integrated servoactuators underwent minor modification to increase their rate capability.

Functionally, the purpose of the VSS is to force the host aircraft to respond to evaluation pilot inputs the same way the simulated aircraft would. Artificial feel systems and simulation controlled displays combine with the VSS to provide the evaluation pilot the real flight motions, accelerations, and handling qualities he would experience if seated in the cockpit of the simulated (modeled) aircraft. Specific features of the VSS based simulation system are: independent control of five degrees of freedom, in-flight model architecture selection, and a fail-safe design strategy. The primary hardware components of the VSS are three Rolm Hawk 32 bit digital computers which host the VSS control laws and simulated aircraft models, and interface with the F-16's digital flight control computer. A response feedback technique is used to implicitly model the desired flight characteristics. The VSS is capable of supporting a model written in any higher order language. Safety features include redundant manual VSS disengage switches, emergency safety pilot override capability, and a system of approximately 100 automatic safety trips keyed to aircraft performance limits and system health.

VISTA has a versatile data acquisition system built around the Air Force Flight Test Center's standard, the Airborne Test Instrumentation System (ATIS). It includes the AR-700 digital data recorder

with 14 track, 90 minute recording capability; video and voice recorders; and triple, secure telemetry data links. The capability for encoding and formatting the flight data into an IRIG-compatible serial bit stream is included.

Provision for future growth is built-in to keep pace with expanding technology over the anticipated 25 year life of VISTA. Space has been set aside in the equipment bays and the dorsal fairing for a reprogrammable display generator, a variable feel sidestick, and the additional computational power required for explicit model following, should that be desired. Approximately 9600 cubic inches of space is also available for customer use as required. Hydraulic lines and wiring harnesses have been sized and routed with an eye to the possibility of additional control surfaces and pods.

FUTURE APPLICATIONS

History has clearly demonstrated a continuing role for in-flight simulation in concert with ground based simulation. In-flight simulation is particularly effective for research into stability and control, and flying qualities. Experience has shown that high workload tasks, in particular, can be more realistically evaluated in-flight. In the past, potentially dangerous control problems have gone undetected even after extensive ground simulation. Those instances serve to remind us that the real world visual and motion cues are key ingredients in the pilot/vehicle dynamics.

VISTA will be applied in three arenas: (1) support to the development and pre-first flight checks of new aircraft; (2) test pilot training; and (3) research on flying qualities, pilot/vehicle interface, flight control systems, and avionic integration issues. The F-16's speed, altitude, and load factor capabilities provide a significantly larger simulation envelope than previously available, enabling VISTA to address the dynamic ranges of current and next generation high performance aircraft. Perhaps equally important, the F-16's modern avionics sensor suite and mux bus architecture will enable VISTA to address the flight vehicle integration issues of combat maneuvering and avionics/flight control system integration.

VISTA is expected to contribute to the Advanced Tactical Fighter program, to attack aircraft

development, to the NASP program, and to the Bomber/Fighter Trainer program. Research areas are expected to include use of vectored thrust, high angle of attack maneuvering, self-designing flight control, minimum flyable dynamics, effects of stick dynamics, use of flat panel displays, helmet-mounted CRTs, pictorial presentations, and other advanced display concepts. A Test Pilot School curriculum with VISTA can feature such topics as the use of radar in fighter air-to-air and air-to-ground missions, pitch pointing, maneuvering using direct lift control, phugoid effects on trimmability, backsidedness, speed stability, steep flight path angle landings, and displays for all-attitude fighter tactics.

VISTA's use will be managed by the USAF's Wright Laboratory. As with the NT-33A and TIFS, actual operation and maintenance will be contracted with an independent firm. Customer projects will be performed on a cost reimbursement basis with full customer task planning, and data gathering and handling. Customer proprietary rights will be protected. Customer cost is made up of a fixed aircraft operational cost plus any additional engineering, data processing, or hardware modification required. This cost will vary from an estimated $5000 per flight hour for simple repetitive training tasks to higher numbers for more complex research tasks. VISTA will eventually be based at the contractor's site but a transportable support system will enable operation from other sites. There will be an initial operational period at Edwards Air Force Base to permit establishment of the required ground support equipment and spares at the operating contractor's site.

SUMMARY

Thirty-four years' experience with the NT-33A has underscored the important role of in-flight simulation. Its contributions to handling qualities research, test pilot training and support of new aircraft development have repaid its cost many times over. More importantly, it has probably saved lives and averted major technological setbacks. Now, due to its age, the NT-33A is near the end of its illustrious road.

The need for in-flight simulation is expected to continue, and even expand, in the future. The Variable Stability In-Flight Simulator Test Aircraft offers a viable high performance replacement for the NT-33A. VISTA's configuration should provide the performance, experimental flexibility, safety features, and growth potential necessary to support the aerospace research and development community for the next 25 years. It will be operationally available by the end of 1992.

Those expected to benefit from VISTA availability run the gamut from the Air Force and Navy Test Pilot Schools, to US and foreign military customers, to worldwide aerospace industry. VISTA is an essential tool for the future.

REFERENCES

1. L.H. Knotts, J.H. Peer, and N.C. Weingarten: "In-Flight Simulation of the JAS-39 Gripen," Society of Experimental Test Pilots 22nd European Symposium, Arles, France, May 1990.

2. G.K. Hellmann, D.E. Frearson, and J. Barry, Jr: "VISTA/F-16: The Next High Performance In-Flight Simulator," AIAA Flight Simulation Technologies Conference, Atlanta GA, September 1988.

3. Capt M.V. Dunbar and G.K. Hellmann: "VISTA/F-16: An In-Flight Research Tool," National Aerospace Electronics Conference, Dayton OH, May 1989.

Safety Cockpit
• Automatic Throttle Tracking
• Config. Control Panel
• Repeater Displays

VSS Signal Conditioning Equipment

CSFDR

AFT Seat HUD Monitor

Dorsal Fairing

In-Flight Refueling Capability

VSS Expansion Chassis (Growth)

Evaluation Cockpit
• Evaluation Displays
• Variable Feel Controllers
• Center Stick Capability

41

Data Acquisition System
• AR 700 Digital Data Recorder
• Video VCR with Split Screen
• IRIG B Time Code Generator
• Telemetry Data Links

VSS Computers
• 3 HAWK-32 Computers
✓ Simulation
✓ Flight Control
✓ Display

Reconfigurable Display Processor (Provision Only)

Modified DFLCC
• VSS Engage/Disengage
• VIM
• Core DFCS Functions

Heavy Weight Landing Gear

Figure 2   VISTA/F-16 Aft Station

42

Figure 3 VISTA/F-16 Forward Station

THE VISTA/F-16 SIMULATION CAPABILITIES

K.S. Govindaraj* , Arvin/Calspan Corporation, Buffalo, N.Y.;
B. Nguyen** , WRDC, Wright-Patterson AFB, Ohio

Abstract

The Variable Stability In-Flight Simulator Test Aircraft (VISTA), a modified F-16, is currently under development by General Dynamics and Arvin/Calspan under contract with the U.S. Air Force. An analysis effort was performed to investigate the VISTA capability to simulate a wide range of dynamic characteristics using two methods; frequency and time domain. The frequency domain method represented the F-16 by a transfer function model. The frequency response of this higher order transfer function was computed for a particular set of feedback gains and matched to a lower order model frequency response. The response match was accomplished using the McFit program. This optimization resulted in a set of lower order model parameters which provided a measure of the closed-loop dynamic characteristics. This technique was used to investigate the range of simulation capability. The time domain method used a simulation software that consisted of the full F-16 non-linear equations of motion. For each feedback gain set, the time response of the VISTA was computed for a command input and matched to the time response of a lower model for the same command input. The response matching was accomplished using a non-linear optimization technique by adjusting the lower order model parameters. The lower order model parameters, which provide a measure of the VISTA closed-loop dynamic characteristics were accepted as valid if certain optimization criteria were satisfied. The analysis indicated that the VISTA will be able to achieve the goal for the simulation capability defined by the Air Force. The actual simulation capability will be determined during the VISTA flight test.

Introduction

The Variable Stability In-Flight Simulator Test Aircraft (VISTA) is presently under development by General Dynamics and Arvin/Calspan under contract with the U.S. Air Force. The VISTA is an F-16D aircraft whose cockpit environment, control feel, and flying characteristics can be changed by the variable stability system (VSS) to match those of another aircraft. It is intended to replace the NT-33A, the Air Force's current high performance in-flight simulator.

The in-flight simulators, both the USAF's NT–33A and the Total In-flight Simulator (TIFS) have been employed extensively to support aerospace research and development effort over the last two to three decades. The NT-33A has been used to simulate nearly every new high performance aircraft developed in the U.S. over the last 20 to 30 years in addition to being used as the primary tool in the development of current flying qualities specifications. A primary mission of the NT-33A is to support the USAF and Navy test pilot schools in the training of test pilots. The VISTA will continue to support test pilot school work, new aircraft development, and flight control/display research. In addition, VISTA will offer significant increase in performance over existing in-flight simulators and offer new in-flight simulation opportunities for avionic integration research and development. A description of the VISTA System and Current Configuration[1] and description of the VISTA Computational capability[2] are provided elsewhere.

The simulation concepts used by the current operational in-flight simulations are: model following and response feedback. The model following simulation concept is used by the TIFS. This in-flight simulator has independent control of all six degrees-of-freedom of motion. The equations of motion and the flight control system of the aircraft to be simulated (model aircraft) are implemented in computers on-board the host aircraft. The evaluation pilot inputs drive the model aircraft control system and equations of motion. The computed model aircraft responses are converted to host aircraft surface commands by the model following control system. These commands force the host aircraft responses to match the model aircraft responses. A simplified block diagram of the model following simulation concept is shown in Fig. 1. The model following control law uses feedforward gains to convert the model responses to host aircraft surface commands. The feedforward gains are dependent on host aircraft aerodynamic parameters. In theory, good model following is achievable with the use of feedforward gains alone. In practice, error feedback (between model and host aircraft responses) is needed because of the lack of exact knowledge of the host aircraft aerodynamic parameters and for supression of the response to gusts.

*Principal Systems Engineer,
 Member AIAA
**Flight Control Engineer

The response feedback simulation concept is used on the NT-33A. This in-flight simulator has independent control over limited degrees-of-freedom of motion. The unaugmented dynamics of the model aircraft are matched by the host aircraft through feedback of appropriate responses. The model aircraft flight control system is closed around the simulated unaugmented model. The host aircraft sensors are transformed to the model aircraft sensor axis system for use by the model control system. A simplified block diagram of the response feedback simulation concept is shown in Fig. 2. The full simulation matches the selected degrees of freedom-of-motion of the model aircraft.

The VISTA has independent control of five degrees of freedom of motion: x-force, lift, pitch, roll and yaw. The current phase of the VISTA development (Phase I) will demonstrate the VISTA simulation capability. The capability of this VISTA configuration, as provided by the variable stability system (VSS), will be demonstrated by the following flight test tasks:

• Simulation of a wide range of dynamic modes.

• Simulation of an existing aircraft with a complex flight control system.

The flight tests will establish range of simulation capability of the dynamic modes - short period, Dutch Roll, Roll/Spiral etc. The X-29A aircraft has been selected as the model aircraft for simulation by the VISTA. This simulation will demonstrate the computational capability of the VISTA on-board computers and the capability of VISTA to simulate aircraft with complex flight control system.
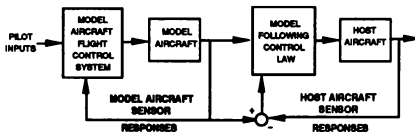
Fig. 1. Model following system.
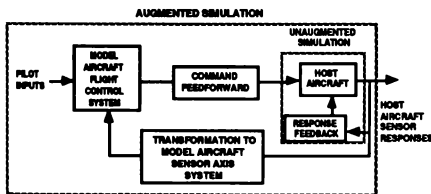
Fig. 2. Response feedback system system.

The capability to simulate a wide range of dynamic modes will be utilized by the Air Force and Navy test pilot schools for pilot training. An analysis effort was performed to determine this simulation capability. The capability to simulate the following dynamic modes was investigated: short period, phugoid, Dutch Roll, Roll/Spiral, n_z/α, and ϕ/β. Two approaches were used for this analysis effort: a time domain approach and a frequency domain approach. The remainder of this paper gives a description of the analysis approaches and the results.

VISTA Variable Stability Control System

The VISTA variable stability system provides the control laws for simulating a wide range of dynamic characteristics. The VSS control laws are implemented in the VISTA on-board HAWK 32 Computers. The VSS control system consists of three major blocks:

• Flight control system of the airplane to be simulated (model FCS)

• Command feedforward system (CF)

• Response feedback system (RF)

Model Flight Control System

The flight control system implemented is specific to the aircraft being simulated. The VISTA variable feel system provides the desired model aircraft feel system characteristics. The evaluation pilot's control commands are the outputs of the feel system and the inputs to the model flight control system. The outputs of the model flight control system are the actuator commands of the aircraft being simulated. These actuator commands are converted to acceleration commands and passed to the command feedforward system which converts these accelerations to VISTA actuator commands. The X-29A has been selected as the model aircraft to be simulated during VISTA flight test.

For test pilot school configurations, a flight control system that demonstrates various dynamic characteristics has been implemented. The pilot command inputs (force or position) are passed through pure time delay, dead band, non-linear gradient and lead/lag filter to simulate various command characteristics. Trim increment to the command control is provided in all three axes. The command path characteristics are adjustable during flight. The desired closed loop dynamic characteristics are provided by the VSS feedback loops. The pitch and throttle command control systems are shown in Figs. 3 and 4, respectively. The lateral and directional command control systems are shown in Figs. 5 and 6, respectively.

The flight control system block also includes a flexible flight control system in the pitch axis that can be varied for research use or as a Test Pilot School project. A block diagram of this flight control system is shown in Fig. 7. The command path includes pure time delay, dead band, non-linear gradient, and lead/lag filter. A trim increment to the command signal is also provided. The command path has the option to use either the force or the position input. The outer feedback loop consists of feedback from of, Δn_z, α, and proportional plus integral control. The inner loop provides α and q feedback. An option is provided in the feedback path for a lead/lag instead of the proportional plus integral control. A second order lead/lag filter on the total command provides the capability to simulate various actuator dynamics. The control system gains and filter parameters are adjustable during flight. The actuator commands computed by the flight control system are converted to acceleration commands by appropriate gain multiplication. The control system provides a wide range of control characteristics that can be used for research or demonstration purposes.

Fig. 5. Lateral command control system.
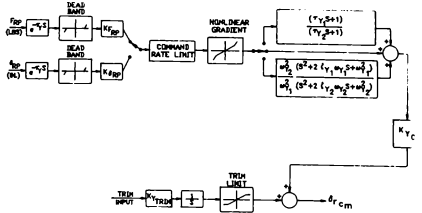
Fig. 6. Directional command control system.

Command Feedforward System (CFF)

The command feedforward system converts the model acceleration commands to the VISTA actuator commands through appropriate gain multiplications. The feedforward gains are a function of the VISTA control derivatives. The actuator commands computed by the CFF represents part of the VISTA actuator commands needed to simulate a model aircraft. The CFF for the longitudinal and lateral/directional axes are shown in Figs. 8 and 9, respectively. For TPS type control system, the CFF commands include the effects of command path characteristics or the outer loop control law. For model aircraft simulation, the CFF commands contain the effects of the model control system.

Response Feedback System (RF)

The response feedback system provides feedback commands to the VISTA actuators to simulate the desired closed-loop dynamic characteristics. The longitudinal and the lateral/directional response feedback systems are shown in Figs. 10 and 11, respectively. The figures show quite a few responses as feedback. Normally, the following signals are used for feedback: α_{CF}, q, p, r, β_{CF}, $\dot{\alpha}_{CF}$, $\dot{\beta}_{CF}$, U_g, α_g and β_q. Some of the TPS configurations require high feedback gains on the angle of attack and sideslip sensor signals. Since these sensors contain turbulence effects, they are

Fig. 3. Pitch command control system.

Fig. 4. Throttle command control system.

Fig. 7. Test pilot school pitch flight control system.

Fig. 8. Longitudinal command feed forward system.

Fig. 9. Lateral/directional command feed forward system.

47
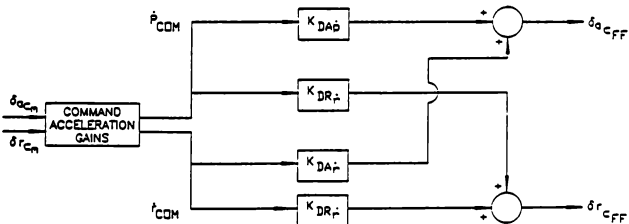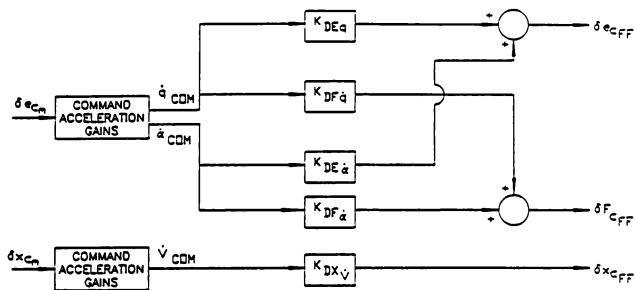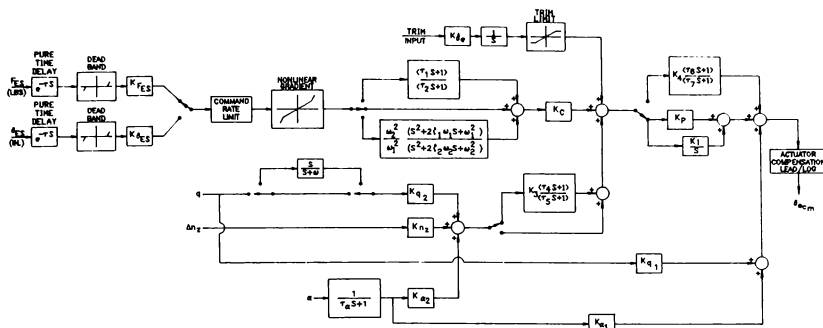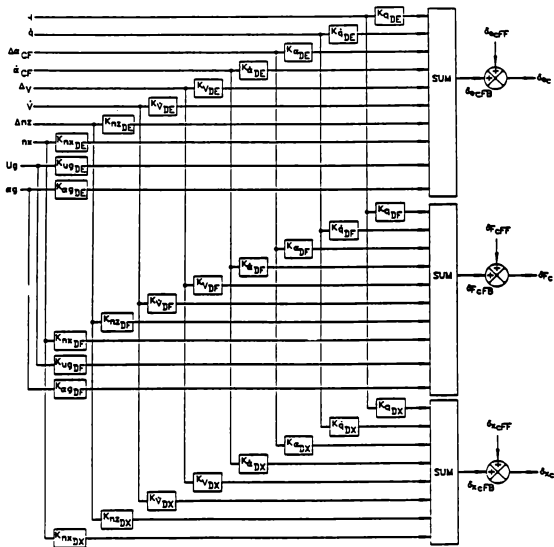
Fig. 10. Longitudinal response feedback system.
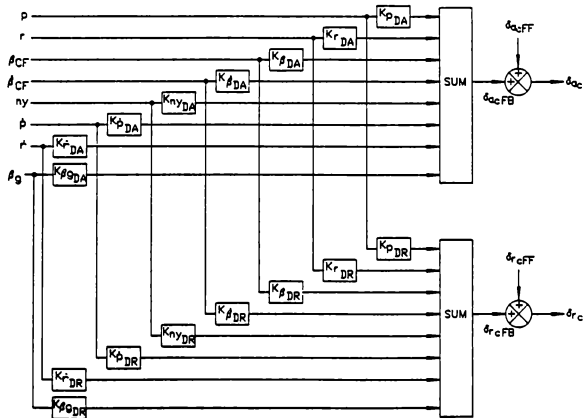
Fig. 11. Lateral/directional response feedback system.

48

not used directly in the feedback loop. The angle of attack and the sideslip signals are complementary filtered with inertial measurements to generate smoothed signals, α_{CF} and β_{CF}, which are used in the feedback loops. The complementary filters combine the airmass related angle of attack and sideslip signals with the corresponding rate signals derived from several inertial signals to generate the smoothed signals α_{CF} and β_{CF} and their rates $\dot{\alpha}_{CF}$ and $\dot{\beta}_{CF}$. These filtered signals have the desired low and high frequency characteristics, are immune from noise and turbulence, and are suitable for use by the response feedback system with high feedback gains. The total VSS commands are obtained by summing the command feedforward and response feedback signals.

The total VSS commands are combined using the F-16 control mixer logic to produce the individual surface commands as shown in Fig. 12. The total VSS provides a five degree-of-freedom simulation. An analysis effort was performed to determine the range of dynamic characteristics that can be simulated by the VISTA using the response feedback gains. This analysis was performed using two approaches – time domain and frequency domain. Calspan used the time domain approach and the VISTA ADPO used the frequency domain approach. A description of the two analysis approaches and the analysis results are given in the following two sections.

Fig. 12. Control surface mixer.

Time Domain Analysis of VISTA Simulation Capability

The analysis effort focused on determining the range of dynamic characteristics that the VISTA can simulate. As stated previously, the capability to simulate a wide range of the following dynamic modes was investigated: short period, phugoid, Dutch Roll, Roll/Spiral, n_z/α and ϕ/β. The analysis was performed for two flight conditions – cruise and landing approach. To support this analysis effort, six-degree-of-freedom simulation software was developed. The F-16 simulation data package supplied by General Dynamics was incorporated into this software. It includes the following: aerodynamic force and moment coefficients computed from table look up data, engine model, fourth order actuator model with position and rate limits, the F-16 digital flight control system, VSS Control law and pure time delay on the actuator commands to simulate computational time

delay and filter time lags. The simulation software does not include the structural mode effects.

The capability of VISTA to simulate a wide range of dynamic characteristics is subject to certain practical limitations due to aircraft mass and inertia, available control power, actuator bandwidth, computational time delays, structural modes etc. Time response analysis technique was used to determine the VISTA simulation limits. The analysis procedure is outlined as follows:

• The response feedback gains required to simulate the various dynamic characteristics were computed using linearized aerodynamic models.

• Time responses for each gain set were computed using the non-linear six-degree-of-freedom F-16 model. A step command input was used to generate the time responses. A block diagram of this simulation is shown in Fig. 13.
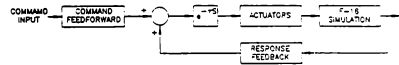
Fig. 13 VSS time response simulation.

• The closed-loop VISTA/VSS system (higher order system) was matched to an equivalent lower order model. Two types of lower order models were used - first and second order. These are defined as follows:

First order: $\quad \dfrac{Ke^{-\tau_e s}}{\tau s + 1} \quad\quad (1)$

Second order: $\quad \dfrac{Ke^{-\tau_e s}}{s^2 + 2\zeta\omega s + \omega^2} \quad\quad (2)$

The matching of the VISTA with the lower order model was accomplished by comparing the time history from the VISTA nonlinear simulation to the time history generated by the lower order model for the same command input. A block diagram of this equivalent model is shown in Fig. 14. The lower order model provides a measure of the VISTA closed-loop dynamic characteristics. The lower order model type was based on the particular VISTA response selected for matching.
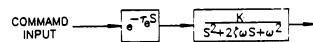
Fig. 14. Equivalent model to match the VISTA closed-loop system.

49

- The response matching was accomplished by adjusting the lower order model parameters to minimize the error between the two responses. The criterion used for optimizing the lower order model parameters is defined as:

$$\text{Error} = \frac{1}{T_2 - T_1} \int_{T_1}^{T_2} |X_{model} - X_{VISTA}| \, dt \quad (3)$$

where X_{model} is the lower order response, and T_1 and T_2 are the start and final times of the time response simulation, respectively. A particular VSS gain set and the corresponding closed-loop dynamic characteristics were accepted if a model set of parameters minimizing the error satisfied the following:

1. $$\text{Min Error} < 0.15 \frac{1}{T_2 - T_1} \int_{T_1}^{T_2} |X_{model}| \, dt \quad (4)$$

2. The VISTA response and the model response are within ±.07 seconds at peaks, valleys and zero crossings.

3. The peak-to-peak magnitudes of the VISTA response correspond with the peak-to-peak magnitudes of the model response within ±15% for commands from D.C. to frequency of up to 2 Hz.

The time response matching of the lower order model response to the VISTA response was accomplished through non-linear optimization. The time response of the lower order model was computed by expressing the transfer function in state variable form and solving the state equations using a fourth-order Range-Kutta integration technique. The error criterion of Eq. (3) was minimized using the non-linear optimization technique of Fletcher and Powell[3]. The error function of Eq. (3) and the three acceptance criteria are as specified in the Air Force statement of work.

The two flight conditions for which the analysis was performed are summarized as follows:

Cruise:

V_{EQS} = 350 Kts
M = 0.78
h = 20,000 feet
Gear and flaps up

Landing Approach:
V_{EQS} = 160 Kts
M = 0.25
h = 3200 feet
Gear down
Flaps down 10°

The two primary effects considered on the VISTA simulation capability were the computational capability and the existing bandwidth of the F-16

integrated servo actuators. The computational time delay was approximated by 18 msec of pure time delay on commands to the actuator. The full actuator model supplied by General Dynamics was incorporated into the simulation software used for analysis. This model is represented by a fourth order transfer function. For response matching using the lower order model, the equivalent time delay was held fixed at 100 msec. The time delay due to actuators was estimated to be 82 msec and the additional computational time delay of 18 msec provided an estimate of 100 msec for the lower order model equivalent time delay. Although the lower order model equivalent time delay could have been used as a parameter optimization, this value provided the best results and reduced the complexity of the response matching process.

The Air Force statement of work contained a goal for the range of dynamic modes to be simulated. This goal was defined for a more advanced VISTA with 10 htz control surface actuator. It is used in the current program as a yardstick for performance with the existing 3 htz actuators. For each dynamic mode, the goal was specified in terms of a range of damping and frequency. The goal for each dynamic mode is shown graphically in Figs. 15-18. These figures show both complex and real pairs of roots. The scale across the bottom is $2\zeta\omega_n$ or $\lambda_1 + \lambda_2$, where λ_1 and λ_2 are negative of the real roots. The scale on the vertical right hand side is ω_n^2 or $\lambda_1\lambda_2$. The scale on the vertical left hand side is ω_n or square root of $\lambda_1\lambda_2$. The parabolic lines represent lines of constant damping ratio. The statement of work did not require that this goal be achieved. The purpose of this analysis effort was to determine if the goal can be achieved, and if not achievable to determine the limits of the dynamic modes that can be simulated. A VSS configuration that satisfied the three previously stated criteria was accepted as a valid configuration. For each dynamic mode, gain sets were analyzed till a particular set just failed to meet the three criteria. The dynamic characteristics corresponding to this set was considered as the simulation limit.

The analysis indicated that the goal was achievable for all the dynamic modes except the short period mode for the cruise flight condition. As the VSS integration with the General Dynamics hot bench facility progressed, the simulation software was modified to reflect the different computational rates. These computational rates are summarized as follows: VSS Hawk Computers – 16 msec, F-16 ground simulation computer – 10 msec, the F-16 actuator model simulation computer – 2.5 msec. This modification of the analysis software to reflect the staggered computations of the hotbench had minimal impact on the original analysis results. In order to achieve the goal for the short period simulation capability for the cruise flight condition, a lead compensation on the VSS

actuator commands was incorporated. This lead compensation effectively increases the actuator bandwidth within the rate limit and other physical actuator constraints. Analysis with this lead compensation included indicated that the goal for the short period for the cruise condition is achievable. The analysis results are summarized in this section.

Short Period

The goal for the short period damping and frequency for the cruise and landing approach flight conditions is shown in Fig. 15. The combination of damping and frequency for the cruise condition is shown by the solid line and for the landing approach condition by the dashed line. The dynamic range of short period damping and frequency without actuator lead compensation is indicated by $\square$ and with lead compensation by O for the cruise condition. A combination of feedback from $\alpha_{CF}, \dot{\alpha}_{CF}$ and q was used to achieve the combination of closed-loop damping and frequency. The VISTA angle-of-attack response for a pitch stick step input was generated and matched to the second order model response generated for the same stick input. The angle-of-attack response best approximates the second order model response and provides an accurate estimate of damping and frequency. The lower order model damping, frequency and gain were adjusted during the optimization to match the VISTA angle-of-attack response. The lower order model equivalent time delay was held fixed at 100 msec. The particular VSS gain set and the resulting closed-loop dynamics accepted as valid if the matching criteria (Eq. 3 and criteria 1-3) were satisfied. The cruise configurations shown in Fig. 15, both with lead compensation and without lead compensation, satisfied the criteria. For the landing approach flight condition, actuator lead compensation was not necessary to achieve the goal for the short period damping and frequency. Time history match between the VISTA and the lower order model response is shown for one of the cruise configurations in Fig. 15.
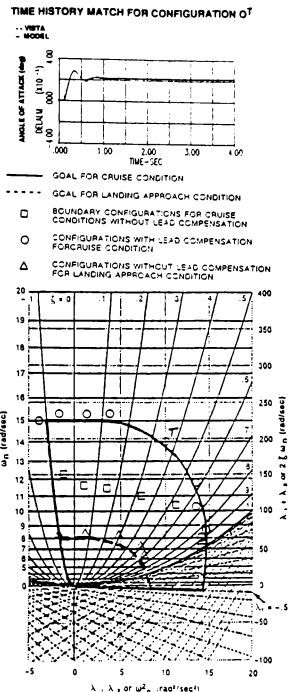
Fig. 15. Short period damping and frequency range.

Dutch Roll

The goal for the Dutch Roll damping and frequency for the cruise and landing approach flight conditions is shown in Fig. 16. The VISTA sideslip response to a rudder pedal input was used to match the second order model response. The lower order model damping, frequency and gain were adjusted during response matching. A combination of roll rate, yaw rate, complementary filtered sideslip and sideslip rate signals were used as feedback to the aileron and the rudder to achieve the closed-loop Dutch Roll damping and frequency. The particular VSS gain set and the corresponding damping and frequency were accepted as valid if the optimization criteria were satisfied. The analysis configurations are shown in Fig. 16 for both flight conditions. The goal was achieved without the need for actuator lead compensation. Time history match between the VISTA and the lower order model response is for one of the cruise configurations shown in Fig. 16.

GOAL FOR CRUISE FLIGHT CONDITIONS

GOAL FOR LANDING APPROACH FLIGHT CONDITION

○ CRUISE CONFIGURATIONS

□ LANDING APPROACH CONFIGURATIONS
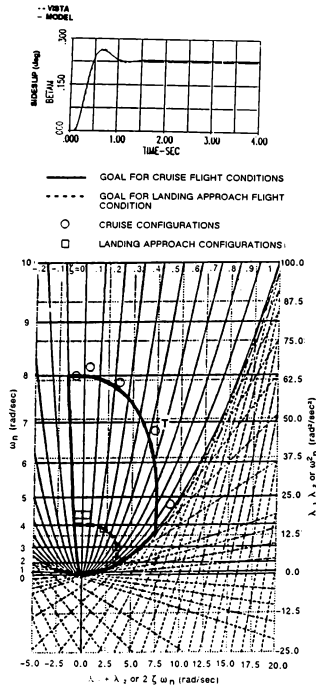
Fig. 16. Dutch roll damping and frequency range.

Roll/Spiral

The goal for roll/spiral modes for the cruise and landing approach flight conditions is shown in Fig. 17. Both the VISTA roll rate and bank angle responses were matched to appropriate lower order model responses to estimate the roll/spiral modes. The VISTA responses were generated for a roll stick command input. A combination of roll rate, yaw rate, complementary filtered sideslip and sideslip rate, and bank angle signals were used as feedback to both the aileron and rudder to achieve the closed-loop roll/spiral modes. For configurations with real roll and spiral roots, the optimization was done in two stages. The short term VISTA roll rate response to roll stick input was used to match the first order model response. The time constant and the gain were adjusted during optimization with the equivalent time delay held fixed. In the second stage, the bank angle response was used to match the second order response. The roll mode was held fixed and the spiral

mode estimated during the second order model fit. For complex roll/spiral modes, the bank angle response was matched to the second order model response to estimate the damping and frequency. As shown in Fig. 17, the goal for the roll/spiral modes was achieved without the need for actuator lead compensation. Time history match between the VISTA and the lower order model response for one of the cruise configurations is shown in Fig. 17.

GOAL FOR CRUISE FLIGHT CONDITIONS

GOAL FOR LANDING APPROACH FLIGHT CONDITION

○ CRUISE CONFIGURATIONS

□ LANDING APPROACH CONFIGURATIONS

Fig. 17 Roll spiral mode range.

Phugoid

The goal for the phugoid damping and frequency for the cruise and landing approach flight conditions is shown in Fig. 18. The phugoid mode can be varied through feedback of velocity and integral of velocity to the throttle or through feedback of velocity to the throttle and horizontal tail. The analysis effort used the feedback of velocity and the integral of velocity to the throttle to vary the phugoid mode. The goal for

phugoid dynamic mode was achieved for both flight conditions, as shown in Fig. 18. The VISTA responses for a throttle step command were generated and the integral of the velocity response was matched to a second order model to estimate the phugoid damping and frequency. The integral of velocity best represents a second order model response. Time history match between the VISTA and the lower order model for one of these configurations is shown in Fig. 18.

TIME HISTORY MATCH FOR CONFIGURATION O^T

GOAL FOR BOTH CRUISE AND LANDING APPROACH FLIGHT CONDITIONS

O   CRUISE CONFIGURATIONS

□   LANDING APPROACH CONFIGURATIONS
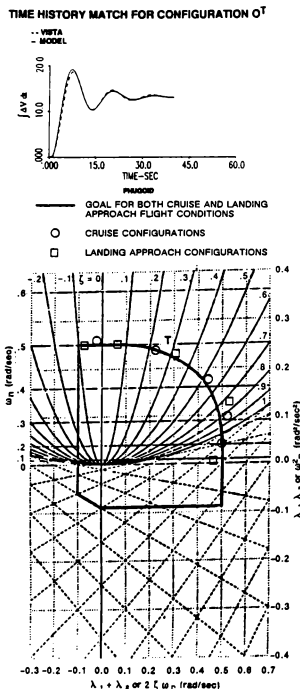
Fig. 18. Phugoid damping and frequency range.

Roll to Sideslip

The goal for the range of ϕ/β magnitude in the Dutch roll mode was defined as $0 \le |\phi/\beta|_{DR} \le 10$. The range of phase of the oscillation in ϕ relative to the oscillation in β was defined to be ±180°. A combination of roll rate, yaw rate, complementary filtered sideslip and sideslip rate signals, as feedback to the aileron and rudder was used to achieve the roll to sideslip goal.

Load Factor Per Angle-of-Attack

The goal for n_z/α was defined in the work statement as $1 \le n_z/\alpha \le 100$. A combination of pitch rate, complementary filtered α and $\dot{\alpha}$ to the horizontal tail and trailing edge flap was used to achieve the goal.

Frequency Domain Analysis of VISTA Simulation Capability

For this analysis, a non-real time, linear, six-degree-of-freedom, High-Order Model (HOM) simulation of the VISTA aircraft using Matrix, a control design and analysis program, was used to determine the frequency response characteristics for a range of feedback gain sets. For each gain set, a frequency-domain matching program, McFit, was used to obtain a LOM. The range of LOM response characteristics were graphically plotted to illustrate the VISTA simulation capability. The analysis was limited to short period mode for the cruise flight condition.

High Order Model

The HOM created with Matrix consisted of linearized F-16 plant dynamics, fourth-order actuator model, and second-order Pado approximation for 18 msec system transport time delay ($e^{-\tau s}$). The 18 msec system transport time delay modelled the Hawk computation. A block diagram of HOM is shown below. This model was validated with a check case from a non-linear six-degree-of-freedom simulation. Time histories of the nonlinear simulation (Calspan's HOM) check case and the corresponding Advanced Development Program Office (ADPO) HOM response are shown in Fig. 19.

LINEAR HIGH ORDER MODEL
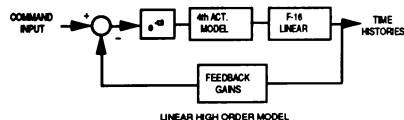
For longitudinal dynamics, three feedback gains were used: angle of attack (K_α), angle of attack rate ($K_{\dot{\alpha}}$) pitch rate (K_q). The dynamics of VISTA can be altered by adjusting the feedback gains. For this analysis the range of values used for each feedback gain was limited to those used by Calspan in their analysis. Different combinations of these feedback gains were set in the HOM in Matrix. For combinations of these feedback gains, the transfer function of the delta elevator command to angle of attack response was determined.

ANGLE OF ATTACK RESPONSE
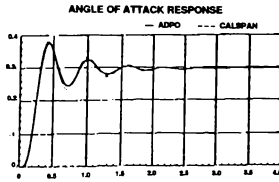
Fig. 19. ADPO and Calspan time history check.

Lower Order Model

Once the dynamic characteristics of the HOM were established for a given set of feedback gains, the McFit program was used to find the LOM which most closely matched the HOM transfer function. The form of the LOM used in this analysis is shown in the block diagram below. The LOM parameters were the equivalent time delay (τ_e), the equivalent damping ratio (ζ), the equivalent frequency (ω), and the equivalent transfer function gain (K). Occasionally, the LOM denominator found by McFit consisted of two real roots instead of a complex pair. In such cases the denominator was described by $(s + \lambda_1)(s + \lambda_2)$.

LINEAR LOWER ORDER MODEL
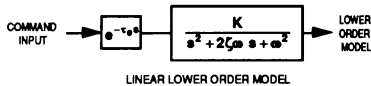
Matching Technique

The McFit program was developed by the McDonnell Aircraft Company to be used to determine equivalent lower order models of highly augmented aircraft. It is frequently used by the Flight Dynamics Directorate and the Aeronautical Systems Division to determine compliance of such aircraft with the US military flying qualities specifications.

McFit will match, in the frequency domain, a high order transfer function or frequency response data with an equivalent reduced order model. It uses a weighted least-square optimal match of a high order frequency response in amplitude and phase angle of up to fifty frequencies in a user-specified frequency range. The program adjusts the parameters of the reduced-order equivalent system in an iterative multi-variable search until a cost function is minimized. The cost function used in McFit is:

$$Cost = (20/n) \sum_{\omega_1}^{\omega_n} \left[\begin{array}{l} \left(gain_{HOM} - gain_{LOM} \right)^2 + .01745 \\ \left(phase_{HOM} - phase_{LOM} \right)^2 \end{array} \right]$$

where: gain is in dB, phase is in degrees ω is the input frequency, n is the number of discrete frequencies.

For this analysis, the LOM was matched to the HOM at 30 frequencies in the frequency range from 0.1 to 15 rad/sec. Guidance in the flying qualities standard, MIL-STD-1797, recommended a frequency range of interest from 0.1 to 10.0 rad/sec unless the equivalent short-period frequency was greater than 10.0 rad/sec. In this analysis, there were several LOMs with equivalent short-period frequencies greater than 10.0 rad/sec. Therefore, 15 rad/sec was used as the upper frequency range limit for all matches done in this analysis. By doing this, the cost function would be the same for all matches. All of the parameters of the LOM were freed. That is, the search routine could adjust all of the parameters, including τ_e, in the optimization of the match.

Analysis Results

The analysis effort concentrated on determining the VISTA short-period simulation boundary in the longitudinal axis for the cruise flight condition. The predicted VISTA short-period simulation boundary is graphically depicted in Fig. 20. The thick lines indicate the boundaries of VISTA short-period simulation capabilities as a function of equivalent time delay. The bottom boundary is the $K_\alpha = 0.0$ limit. The top boundary is the $K_\alpha = 6.0$ limit. The left most boundary is the $K_q = 0.0$ limit. The boundary in the upper left corner is the $K_q = 0.6$ limit. The $\tau_e = 80$ msec line is the second line from the left. Any points on this line can be simulated with no less than 80 msec of equivalent time delay. This will have minimum effect in practice because all modern aircraft have time delay due to actuators and computations from hardware and software. For this flight condition, the VISTA aircraft could not simulate dynamics with less than 70 msec of equivalent time delay. The boundary on the far right is the $\tau_e = 120$ msec line. Of course, the VISTA aircraft can simulate dynamics with higher time delay than 120 msec.

The τ_e boundary lines indicate that it requires a higher equivalent time delay to simulate higher equivalent short-period frequencies or damping ratio. For example, looking at Fig. 20 the approximate equivalent time delay that the VISTA will exhibit in simulating various combinations of damping and frequency can be summarized as follows: 85 msec to simulate an airplane with $\omega = 5$ rad/sec and $\zeta = .2$, 94 msec to simulate $\omega = 5$ rad/sec and $\zeta = .6$, 114 msec to simulate $\omega = 10$ rad/sec and $\zeta = .6$. Using category A flying qualities in the standard handbook, MIL-STD-1797, the τ_e, ω, ζ boundary lines also describe the VISTA simulation capabilities in terms of flying qualities (Levels, 1, 2 and 3) in Fig. 21. Fig. 22 shows the predicted VISTA short-period simulated boundary overlaid with the VISTA goal.
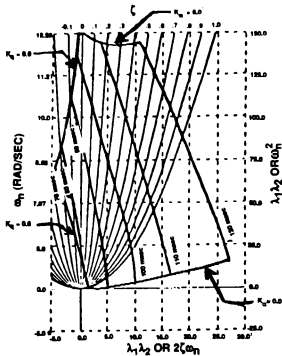
Fig. 20. Short period damping and frequency as a function of equivalent time delay.
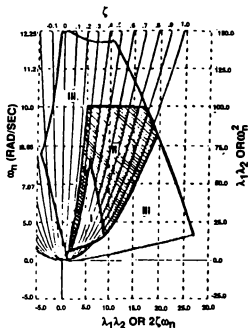
Fig. 21. VISTA simulation capabilities with respect to flying.
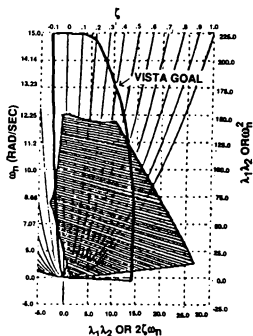
Fig. 22. Short period boundary overlaid with the VISTA goal.

Observations

In the time domain analysis an actuator lead compensator was used to achieve the short period goal for the cruise flight condition. The lower order model time delay of 100 msec used during the optimization produced a good response match between the higher order model and lower order model responses. In the frequency domain, the lead compensator can be incorporated into the higher order transfer function to provide the necessary lead to achieve the short period goal. The impact on the equivalent time delay, included as a parameter in the frequency domain optimization, is unclear till the analysis is performed. The actual VISTA simulation capability boundary will be obtained from flight test.

Conclusion

This paper described the analysis effort to investigate the VISTA simulation capability. Two methods were used for the analysis - time domain and frequency domain, to investigate the range of dynamic characteristics that VISTA will be able to simulate. The Air Force statement of work defined goals for these dynamic characteristics. The analysis indicated that the goal is achievable for all the dynamic modes. For the short period mode in the cruise flight condition, actuator lead compensation was used to achieve the goal.

References

1. "VISTA: An Essential Tool for the Future," D. Frearson, WRDC, Wright-Patterson AFB, Ohio; and P.A. Reynolds, Arvin/Calspan Corporation, Buffalo, N.Y.

2. "Parallel Processors of the VISTA/F-16 In-Flight Simulator," T. L. Landers, Arvin/Calspan Corporation, Buffalo, N.Y.; L. Hamilton.

3. Fletcher, R. and Powell, M.J., "A Rapidly Convergent Descent Method for Minimization." Computer Journal, Vol. 6, No. 2, 1983, pp. 163-168.

PARALLEL PROCESSORS OF THE VISTA/F-16 IN-FLIGHT SIMULATOR

Thomas F. Landers
Calspan Corporation
Buffalo, New York

and

Capt Lynne T. Hamilton-Jones
Wright Laboratory
Wright-Patterson AFB, Ohio

ABSTRACT

This paper discusses the implementation of a multiple processor computer architecture in the Variable Stability In-Flight Simulator Test Aircraft (VISTA). The key to this aircraft's simulation capabilities is a sophisticated Variable Stability System (VSS) comprised of ten central processing units (CPUs), nine of which are programmed mostly in Ada. Efficiency is achieved in the nine Central Processing Units by the use of Direct Memory Access (DMA) interfaces. DMA allows one computer (or interface board) to directly access the memory of another, without involving the CPU in either computer. This feature allows data to be transferred between computers in the system while other computer tasks are being performed. In addition, by using interrupts to control scheduling, a minimum of overhead is needed to implement simple real-time scheduling needed for this application.

BACKGROUND

The Variable Stability In-Flight Simulator Test Aircraft (VISTA) is the next generation high performance US in-flight simulator currently being developed by the US Air Force. VISTA is a modified F-16D in which the cockpit environment, feel system, and flying characteristics can be changed to match a wide array of aircraft. VISTA will be able to replicate the handling qualities of such high performance aircraft as the Advanced Tactical Fighter (ATF), F/A-18, X-30 National Aerospace Plane and many others just by a making entries using the aircraft displays.

A perfect example of a possible VISTA application is the first flight test of the control laws of competitive aircraft designs such as the YF-22 and YF-23. An enormous amount of information on the flying qualities of an aircraft system can be collected before hardware is developed. Any faults in the stability and control characteristics can be exposed safely and corrected. In addition to the application above, the VISTA platform will have other applications: training test pilots in the Air Force and Navy Test Pilot Schools and flight control research on flying qualities. VISTA is scheduled to become operational in the Fall of 1992.

The heart of the VSS is the three ROLM HAWK/32 computers, programmed to use response feedback (RFB) stability augmentation to match the motion of the aircraft being simulated. The VSS computes the model flight control system (FCS) and command feedforward/response feedback control laws. This paper will discuss the VSS functions, hardware, software architecture, and interfaces used to balance the needs of the high update rates required for a high fidelity simulation and the flexibility required to easily accomodate new displays and the new flight control system design.

VSS CONTROL LAWS

The static and dynamic flight characteristics of the VISTA aircraft are altered using the VSS control laws. However, the F-16 control laws remain intact operating independently of the VSS. The Digital Flight Control System (DFCS) provides the safety pilot in the aft seat and the evaluation pilot in the

front seat the option of selecting F-16 control laws or VSS control laws during VSS engagement. The evaluation pilot's commands are the outputs of the VISTA feel system. The pilot's stick inputs are received directly from the centerstick or through the Digital Flight Control Computer (DFLCC) from the sidestick. The rudder pedal inputs come through the DFLCC and the throttle comes in directly via Analog to Digital (A/Ds).

The outputs of this feel system are inputs to the model flight control system as shown in Figures 1 and 2. The model flight control system is the control system of the aircraft being simulated. Two types of flight control systems will be flown during the development flight test program: a typical Test Pilot School configuration and the X-29. The output of the model flight control system is the actuator commands of the aircraft being simulated. These actuator commands are converted to the model acceleration commands, accomplished by multiplying the model control law surface position commands by the gains of the model "B" matrix to produce the desired model aircraft accelerations. These accelerations are passed to the command feedforward system. The inner loop is the Command Feedforward/Response Feedback (CF/RF) portion of the VSS control laws. The inner loop allows VISTA to fly like the open-loop model airframe. The outer loop makes VISTA fly as the closed-loop model airframe by implementing the model flight control laws. The safety pilot can choose the control law elements he wants to use via the VISTA Configuration Control System (CCS). The CCS consists of pages displayed on a Multifunction Display (MFD) where the safety pilot can change every gain in the VSS.

Command Feedforward System

The command feedforward (CF) portion of the inner loop controls the initial response of the aircraft (the location of the zeros). The CF system converts the model aircraft accelerations to the VISTA actuator commands by multiplying the acceleration commands by the gains of the inverse F-16 "B" matrix to produce the F-16 surface actuator commands that will produce the desired model aircraft accelerations. In the longitudinal axis, the pitch acceleration, angle-of-attack rate, and velocity rate commands are converted to the VISTA elevator, symmetrical flap, and throttle commands, respectively. In the lateral/directional axes, the roll

and yaw acceleration commands are converted to the VISTA aileron and rudder commands, respectively.

Response Feedback System

The Response Feedback (RF) section of the inner loop uses the feedback gains from the angle-of-attack, pitch rate, and normal acceleration to change the short period, phugoid frequency, and damping to match the model aircraft. The RF system provides feedback commands to the VISTA actuators to simulate the closed-loop aerodynamic characteristics of the model aircraft. The total VSS commands are obtained by summing the CF commands with the RF commands.

The CF/RF gains are usually fixed during a VSS engagement. The gains are set in an input file before flight as a particular configuration. The safety pilot could change the value of these gains during engagement, but they remain fixed during maneuvers. Also, if the aerodynamics are highly nonlinear functions of flight dynamics, they could be programmed to be scheduled as a function of flight dynamics.

VSS SYSTEM OVERVIEW

Hardware Design

The VSS system consists of eight major components as shown in Figure 3. Each of the Hawk mil-spec computers has a speed of 1.4 million whetstones/sec, 8 MB of random access memory (RAM) expandable to 32 MB, 3.5 MIPS, and is qualified to 9 g's. Each Hawk contains a Network Interface Controller (NIC). The NIC is used for high speed, real time inter-Hawk communication and synchronizing the beginning of the computation frame in each Hawk. In its current use, the transmitting Hawk sets up an output block of data (comprised of Ada records) and issues a start command to the NIC card. The NIC transfers the data directly from the transmitting Hawk's memory to a corresponding data block in the receiving Hawk's memory without involving the CPU of either Hawk. A counter in the last word of the data block allows the receiver to detect the completion of the transfer. When the receiver is ready to use the data, it waits until the transfer has completed, and distributes the data from the contiguous input block to the various records and

arrays. The NIC runs at a 8 MHz rate on a Burst Multiplex Channel (BMC) bus. All information is passed using Ada records.

The Hawk converts Ada arrays and records to representation-specified Ada records. After the conversion to the bus format, the S1000 card is commanded to send out the data. There are five S1000 cards, one in each Hawk and two in the Input/Output (I/O) Expansion Chassis. This card has a 80186-based processor which runs at 8 MHz. The interface between the Hawk and the S1000 cards is a high-level one where Ada data structures are used to pass data between the two CPUs. The VISTA aircraft communication is primarily conducted via the MIL-STD-1553B data buses shown in Figure 4. A special one-block queue in the S1000 is used for textual data sent to the MFDs and special blocks allow Titan software, which controls the variable feel system controllers, to be loaded from the Hawk via the VMUX. The S1000 has Direct Memory Access (DMA) capability which allows it to access the Hawk memory without involving the Hawk CPU. Upon receipt of a transmit command, the S1000 DMAs the output data from the Hawk memory to the S1000 output buffer, which will be picked up by the bus controller on the next 1553B transmit command. When data is received via the 1553B bus, the data is DMA'ed from the S1000 memory to Hawk memory into a "ping-pong" buffer, and the buffer pointer is updated to indicate the new buffer to use. This scheme prevents the Hawk from reading memory that is being written to by the S1000. When the Hawk is ready to use the data, it accesses the proper buffer and converts the data from the 1553B format to floating point or another "natural" data format. Each Hawk contains two A/D cards, 16 channels per card, and one digital to analog (D/A) card. Each A/D card set automatically converts up to 32 analog channels of data and DMAs the integer result to Hawk memory. The two A/D card sets in Hawk #1 and the I/O chassis operate simultaneously. Data is read from and written to discretes via programmed I/O instructions.

Hawk #1 reads the feel system information, computes the model flight control system actuator commands, and records VSS software parameters to the AR700 data recorder. It also has the F-16 simulation for ground simulation. Hawk #1 interfaces to the DMUX, the primary display bus for the F-16 aircraft. This permits VSS simulation

information to be displayed on the F-16 Multifunction Displays (MFDs) and HUD.

The ROLM I/O Expansion Chassis provides extended input/output capability between Hawk #1 and the Airborne Test Instrumentation System (ATIS) via the RMUX, the recording interface, and the Titan computer via the VMUX. The Titan is a 25 MHz, 80386-based computer with a floating point coprocessor. The Titan reads the artificial feel system centerstick information, computes the model feel system responses using model following control laws, and outputs position data to the feel system hardware. It runs on a 4.0 msec cycle time and has a 1553 interface to Hawk #1.

Hawk #2 interfaces to the F-16 avionics bus, AMUX, so that the VSS can be provided with F-16 sensor information, specifically the aircraft position, altitude, velocity, acceleration, and environmental data. Hawk #2 reads the VSS sensors and Inertial Navigation System (INS) information, compares Hawk #2 sensors and Hawk #3 sensors for sensor failures, and computes response feedback actuator commands if dual computations are desired. A special one-block queue is used for textual data sent to the Data Entry/Cockpit Interface Set (DE/CIS) which provides the pilot with data entry capability. The AMUX is similar to the DMUX in operation.

Hawk #3 reads VSS sensors, compares Hawk #2 and Hawk #3 sensors for sensor failure, and computes actuator commands during response feedback operation. Hawk #3 has a dedicated interface to the DFCS via the FMUX. This interface is used to transmit the VSS actuator commands and throttle servo commands from the Hawks to the DFCS. The S1000 card in Hawk #3 converts the floating point data in the control laws to scaled integer data which is used in the DFLCC. After DMAing data to its memory the S1000 sends the surface commands to the DFLCC, retrying if errors occur. Once checked by safety monitors, the commands are sent to the actuators by the DFLCC. This procedure also works the same way in reverse where the DFCS sends data to Hawk #3. The data needed by any other Hawk computer will be transferred from HAWK #3 over the Hawk NIC.

The Engage Logic and Interface Chassis (ELIC) monitors the VSS hardware through dual analog safety trip circuits. It monitors the feel system servos

in the centerstick and throttle for errors. The Hawk and Titan computers send discrete disengage signals from their internal monitors as well as watchdog timer discretes. These signals are "or'ed" together so that any one trip signal will disengage the VSS.

The sensor Signal Conditioning Chassis (SCC) provides the Hawk computers with interfaces to the VISTA analog sensors by scaling and filtering the AC and DC signals to the +/- 10 volts range. The SCC includes a patch panel for flexibility and the capability to test each channel during a self-test.

The Raymond Disk system is the primary load device for the Hawk and Titan computers. It contains a 40 Mbyte disk and can operate up to 9 g's. The Raymond Disk is located in the front cockpit and saves the configuration changes made in flight.

Software Functionality

The VSS software is presently composed of 84K lines of Ada code. The only non-Ada code is a small amount of low-level device drivers written in Assembly. For development flight test purposes, the X-29 was chosen as a complex aerodynamic and flight control system (FCS) model for simulation fidelity verification. The X-29 model FCS is written in Fortran and is the only other non-Ada code presently in VISTA.

The projected timing of the VSS software is shown in Figure 5. A detailed explanation of this diagram follows. Hawk #1's computation frame is started by a timer interrupt. The interrupt service routine (written in Assembly) calls the first block of Ada computations in Hawk #1 (shown in Figure 5 above the checkered region). First, a message is passed to Hawks #2 and #3 on the NIC that generates an interrupt in each of their processors and starts their computation frames (similar to how the timer interrupt started Hawk #1's computation frame). Hawk #1 then commands the VMUX S1000 card to transfer data from the Titan to the Hawk (not shown on the timing diagram since it is a very short event). Hawk #1 then reads its A/D converters and scales the integer values read in by the device to the appropriate floating point values (in engineering units). Hawk #1 then scales the integer values read in by the VMUX (which were DMA'ed into Hawk memory by the S1000 card while the A/D conversion is occuring). The redundant A/D data is then averaged for control

law computations and differenced for safety trip computations. At this point the Ada procedure ends and the timer interrupt exists. The main-loop computations in Hawk #1 now continue (the main loop is what was interrupted by the timer interrupt). Hawk #1's main-loop is computations for the primary pilot interface, the CCS.

Meanwhile, over in Hawk #3, the FMUX S1000 card is commanded to transfer data from the DFLCC to the Hawk on the FMUX (similar to how Titan data was read in on the VMUX in Hawk #1). A/D data is read (similar to Hawk #1) and the data is scaled to engineering units. This data is then transmitted over the NIC to Hawks #2 and #3 then waits for the transmission of A/D data from Hawk #2. In Hawk #2, A/D data is read in and scaled; and then the AMUX data blocks are checked for any fresh data (this differs from the VMUX and FMUX data that was read in Hawks #1 and #3 because Hawk #2 is a remote terminal on the AMUX, whereas Hawks #1 and #3 are bus controllers on the VMUX and FMUX). The latest AMUX data, along with A/D data in Hawk #2 is sent over the NIC to Hawk #3.

Hawk #2 and #3 now proceed in parallel, receiving the data transferred over the NIC ("receiving data" over the NIC is the nontrivial task of taking the data from the contiguous block of Ada records transferred into the receiver's memory and distributing it to the various output Ada records), computing the average of the redundant VSS sensors for use in computations, and computing the difference for use in safety trips. The two Hawks then send the average data to Hawk #1 over the NIC. Hawk #2's transfer also generates an interrupt to the Hawk #1 CPU, which interrupts the main-loop and begins the remainder of the real-time processing in Hawk #1. Hawk #2 has completed the first part of its real-time processing, so the interrupt exits and the CPU becomes idle (there is no useful work currently being done in Hawk #2's main-loop). Hawk #3 completes the first part of its real-time computations by running the complementary filter and response feedback subroutines. It then exits its interrupt, and the Hawk #3 CPU becomes idle.

Hawk #1, now awakened by the NIC interrrupt generated by Hawk #2's NIC transfer, receives the NIC data from Hawks #2 and #3, and calls the model FCS computations. The outputs of the model FCS are then sent back over the NIC to Hawks #2 and #3,

and the transfers again generate an interrupt in Hawks #2 and #3 to finish their real-time computation frames. Hawk #1 continues its processing by calling the remainder of computations needed for various other functions including the real-time pilot interface computations, checking the safety trip status, running the mode control logic, reading and writing its discrete inputs and outputs, outputting data to the RMUX S1000 card for recording, collecting new outputs to be recorded, computing any tests inputs, and ground simulation computations. After all that, the real-time computation frame is complete and Hawk #1 becomes idle (here, an empty loop is used to keep the CPU from doing any useful work -- this is required to meet the spare time requirement for the development of the software).

Notice that the RMUX data is sent out before new data is collected for recording in Hawk #1. This is so all data sent on the RMUX arrives from the same VSS frame (the data from Hawks #2 and #3 arrives one frame late).

Over in Hawk #2, the NIC transfer interrupt "woke it up" and its real-time computations are completed. The NIC data is received, mode control computation are done, discrete inputs and outputs are read from and written to, and data to be recorded is scaled. Hawk #2 has finished its real-time computations, so the interrupt exits and the CPU becomes idle.

In Hawk #3, the NIC transfer from Hawk #1 has generated an interrupt there also. It starts by receiving the NIC data from Hawk #1, running command feedforward computations, "mixing" the actuator commands and balancing them for output to the VISTA surfaces. The actual surface commands are then generated. A timer interrupt is pre-programmed to occur shortly after this, which scales the data for output on the FMUX and then notifies the S1000 to send the data (a timer interrupt is used to reduce the time jitter of the transmission, a requirement of the DFLCC). The FMUX S1000 DMAs the data from the Hawk memory to its own and transmits the data on the FMUX (retrying if errors occur). In the Hawk CPU, while the S1000 is transmitting data on the FMUX, the throttle computations are done and the throttle command is D/A'd. Following that, the safety trip status is checked, mode control computations are performed, subliminal Built-in-Test (BIT) computations are

checked, discrete inputs and outputs are serviced, and data to be recorded is scaled. At this point Hawk #3's real-time computations are complete, the NIC interrupt exits and the CPU becomes idle until the next frame begins.

SENSORS

The sensor/signal conditioning system provides the VSS Hawk computers with interfaces to the analog signals of the VISTA aircraft sensors. It also provides the VSS with analog recording of these sensor signals. There are three types of interfaces to the Hawk computers:

1) Digital from the source via a 1553B bus input
2) Analog input to the Signal Conditioning Chassis, conditioned within the SCC, and then input to the Hawk computer through an A/D
3) Analog direct from the source to the Hawk through an A/D

Most of the VSS sensor signals have dual sources. Some originate from the F-16 sensors, but most from the dedicated VSS sensors. Figure 6 illustrates how the analog sensor inputs flow into Hawk computers #2 and #3. The sensor system is functionally and physically divided into two parts, Channel #1 and Channel #2. Duality is shown by having two power supplies, separate connectors for Channel #1 and #2 signals to enter and leave the chassis, and Channel #1 and #2 sensor conditioning performed on separate circuit cards.

Since linear accelerometer signals can be dominant response feedback signals, dedicated dual analog signals are required. These are linear signals from the aircraft center of gravity and from the pilot's station, each dual redundant in three orthogonal axes (Nx, Ny, Nz). The VSS also requires aircraft angular accelerations (p, q, r) which are calculated from Nx, Ny, and Nz. For the same reasons, as the accelerometers, the angular rate signals will be from dedicated redundant sensors.

There are two cones each for angle-of-attack (AOA) and angle-of-sideslip (AOS). However, the AOS cones are not considered dual sources because a single cone will not remain accurate over the full range. The cones are blended together in the Hawk computer to form a single signal. The aircraft

Inertial Navigation System is the sole source of the attitude signals on the VISTA aircraft.

Figure 7 shows how the digital signals enter the Hawk computers. Hawk computers #2 and #3 perform complimentary filtering of the sensor signals. Using these complimentary filter outputs and the body axis measurements, the following inertial rate signals are computed: vertical acceleration, the rates of change of AOA, AOS, and velocity. The inertial rate signals and sensor signals are blended (complementary filtered) to produce signals that have immunity from noise and turbulence. These signals also become suitable for use by the response feedback system with high feedback gains. High feedback gains may be required on these signals to simulate specific VSS configurations.

SUMMARY

The VISTA is a new high performance in-flight simulator whose on-board simulation system, VSS, computes the command feedforward/ response feedback control laws to match the motion of the aircraft being simulated. The VSS programming is written mostly in Ada, uses DMA interfaces and interrupt-based scheduling. This allows direct access between computer memories while other tasks are being performed in parallel. The actual efficiency and flexibility of this design will be tested during the VISTA flight scheduled to begin November 1991.

REFERENCES

[1] M. V. Dunbar, J. L. Dargan, 'VISTA/F-16 Design Features,' AIAA Flight Simulation Technologies Conference, 14-16 August 1989.

[2] General Dynamics, 'Software Requirements Specification for the Variable Stability System (VSS) of the Operational Flight Program (OFP) of the VISTA/F-16,' Revision A, 27 April 1990.

[3] General Dynamics, 'VISTA/F-16 VSS Design Analysis Report,' 24 January 1990.

[4] General Dynamics, 'Preliminary Design Review Viewgraphs,' VISTA PDR, April 1989.
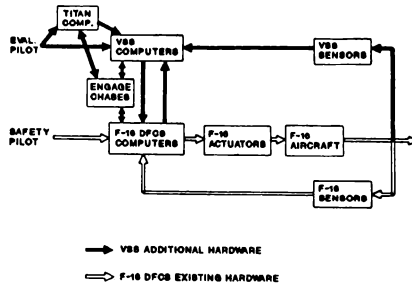
Figure 1: VISTA Architecture
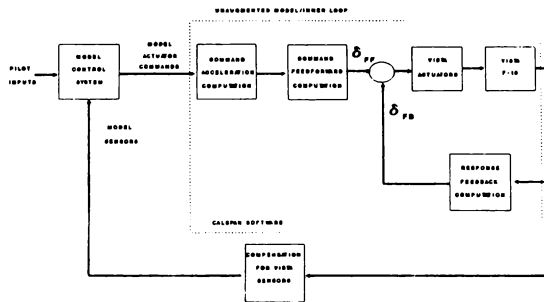
Figure 2: VISTA Simulation Control Laws
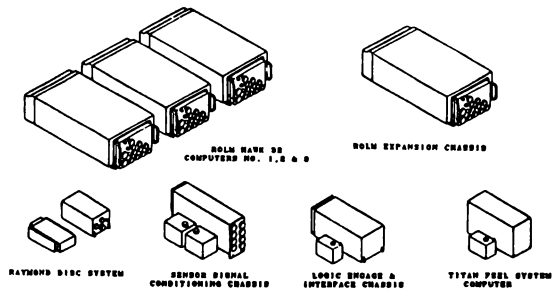
Figure 3: Major VSS Components
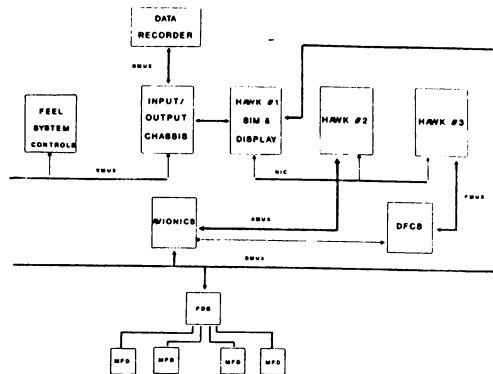
Figure 4: Aircraft Communications
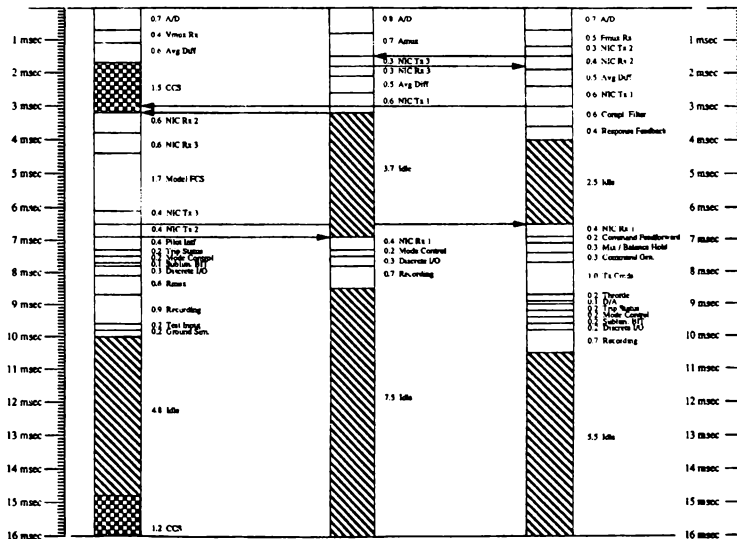
Figure 5: Projected Timing Diagram

63
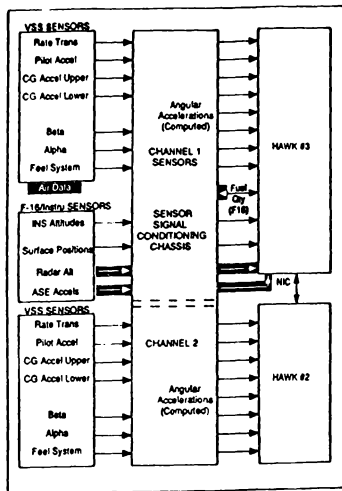
Figure 6: Analog Signal Sensor Flow
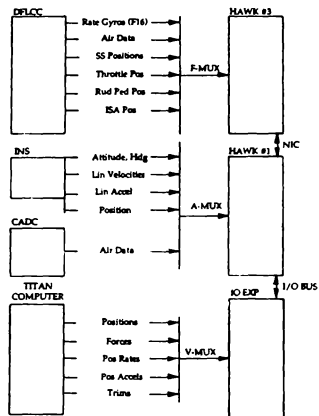
Figure 7: Digital Signal Sensor Flow

THE COMPROMISE BETWEEN ACCURACY AND REALISM IN FLIGHT SIMULATION.

A G Barnes *
Lytham St Annes
Lancashire
UK

Abstract

The usual classification of flight simulators differentiates between simulators for **Research** and simulators for **Training**. Next, the distinction is made between those for Civil use and those for Military use. Further subdivisions follow, based on type of aircraft, and on the operational use of the simulated aircraft.

The paper discusses an alternative way of classifying simulators, based on the interactions between the device and the pilot or crew. First, a distinction is made between simulators which are used to **teach** and to educate, and those which are used to **train** motor skills and coordination. The key feature of a teaching device is **accuracy**; the key feature of a training device is **realism**. This implies that accuracy is less important for the training of motor skills. It is shown that creating an illusion of reality often conflicts with a requirement for accuracy.

The approach described in the paper can be used to shed light on many issues relating to technology, such as the merits of simulators for research on flying qualities, the likely effectiveness of networked simulators, the features needed to achieve virtual reality, and the likelihood of a simulator inducing sickness. The classification also raises the issue of whether today's simulators are over-specified in some respects, and under-specified in others.

1. Introduction.

The distinction between flight simulators designed for pilot training, and those designed for research and development, is well established. Each class of simulator is distinguished in terms of the equipment standard, the utilisation, and the method of operation. They can be further sub-divided on the basis of specific application. For example, training simulators divide into civil or military, specific aircraft type, and the special areas of training - initial, conversion, continuation, operational and full mission (figure 1). From these sub-divisions, the requirements for a new facility or a new simulator can be derived, and later transposed into a specification, based on the appropriate technology.

The element common to all flight simulators is the pilot, or crew. The successful simulator is one in which the desired inter-action between pilot and the device occurs. The purpose of this paper is to propose a classification of simulators based on these inter-actions, to supplement the more traditional breakdown described above. The new classification affords the opportunity to rationalise the solution to a requirement for a new device. It sheds light on why some existing

* Consulting Engineer
  Member AIAA, Fellow RAeS

simulators fall short of expectations. It also assists in predicting whether or not a new configuration or technology is likely to succeed.

2. Basis of the Classification.

To predict the flight path of a missile, scientific principles from the domain of mathematics and physics must be applied. This is a cerebral activity, with success depending on the assumptions made and the accuracy of the calculation. In contrast, when a dog catches a ball, Newton's Laws of Motion are not being solved iteratively in its brain. The process stems from a conditional reflex, an a animal instinct which can be reinforced by practice. Both processes can be improved by tuition.

The two methods of tuition are 1) **teaching** of intellectual skills, and 2) **training**, of motor skills. Relating this breakdown to the control of complex vehicles, and in particular, aircraft, both methods of instruction are involved. Initially, pilot instruction is a classroom activity. Aviators need a knowledge of aerodynamics, engines, systems, navigation, meteorology, and other subjects. In both commercial and military flying, they must acquire and maintain academic skills. Teaching of these skills can be enhanced by aids - books, graphics, work-stations, and other inter-active devices.

Figure 1

CLASSIFICATION BY TYPE

TRAINING — Military (Desk Top, Procedures/Emergencies, OFT/FMS), Civil (Type Trainers, Classroom, Phase 2/3)

RESEARCH & DEVELOPMENT — Government (Future Concepts, Procurement Support, Human Factors), Industry (Technology, Design Support, Project Support)

The task of controlling flight path comes into the second category. Student pilots are taught by example – use of stick and throttle for straight and level flight; judging position on the landing approach; the use of controls in the landing flare. Skill is developed by practice on the machine itself. The theory of flight will help the pilot to appreciate the physical limitations he will encounter, but it will not teach motor skills. In spite of extensive research in control theory and human factors, many gaps remain in our understanding of the relative contribution of visual, motion, and proprioceptive cues.

The distinction between cerebral and motor skills applies both to training simulators, and to simulators used for research and development. The common element is the pilot, and, depending on the area of research, a decision can be made on the type of simulator which is most appropriate. If the research is concerned mainly with mental activity, such as operating a radar display, it will be in the first category. If the research is directed to flight control, or to tasks where a realistic workload is important, then the simulator will be in the second category.

Figure 2 shows the breakdown which results from a classification based on these categories. The features which relate to them will next be discussed.

CLASSIFICATION BY PURPOSE

Figure 2

Category A — TEACHING – Cerebral Skills (Understanding, Investigating, Learning, Optimising, Requirement – Accuracy)

Category B — TRAINING – Motor Skills (Co-ordination, Re-action, Perception, Stress, Requirement – Realism)
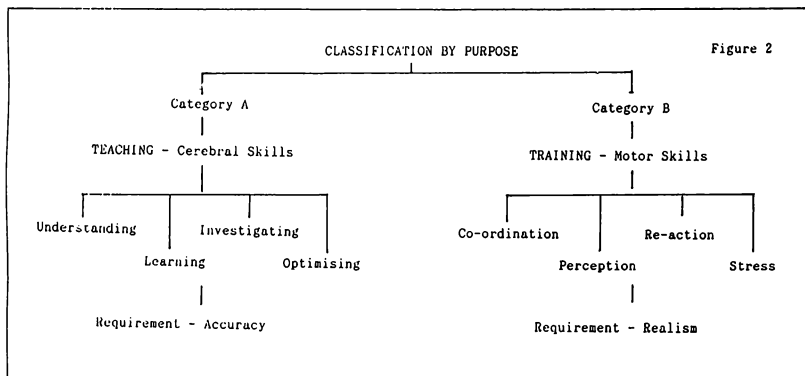
3. Category A. Teaching of Intellectual Skills.

The over-riding requirement for the category A simulator is accuracy. If the information in text books or other storage devices is erroneous, they have little value, Pupils also expect that information from teachers is factually correct. deception does not play a part. Similarly, a Category A device which gives incorrect information is a potential threat to safety. It is also important to know the boundaries within which this class of simulator is accurate, and to design the program of instruction accordingly. There can be less emphasis on reality: in fact, learning will proceed more quickly if the distractions and discomforts of the airborne environment are absent.

Realism has much less importance in this class of simulator. There still has to be a valid relationship between the device and the system which is simulated, but face validity is not a criterion. The primary purpose is to improve the crews' knowledge of the aircraft, its systems, and the operating environment.

The FAA Flight Training Device designations (ref. 1) fall into this category. They include computer based trainers (CBT), cockpit procedure trainers (CPT), cockpit system simulators (CSS), and advanced training devices (ATD). The military equivalents include weapon system trainers (WST), and cockpit and emergency procedures trainers (CEPT). In this class of simulator, the designer is able to use the most appropriate format for instruction, such as inside-out displays, compressed timescales, and an open layout that allows better monitoring by the instructor.

4. Category B. Training in Motor Skills.

Military and Civil Aircraft Operators recognise that a significant proportion of aircrew training can be allocated to ground based simulators. The benefits are proportional to the cost and complexity of the aircraft they operate. Flight simulators have been built for most of the aircraft in operation today.

The ground based simulators used by both the Civil and Military for a significant proportion of their training come into this Category. Although the configuration of these simulators vary, the intention is the same - to reproduce as closely as possible the situation experienced by the crew when performing critical tasks. The region of validity depends on the Specification agreed with the supplier. In the case of the major airlines, they require a simulator which reproduces the complete flight, from start to shut-down, including the ability to inject emergency situations. Consequently, proof of pilot proficiency is largely a ground-based activity.

The user of this class of simulator expects realism. The standard of the cockpit and its furnishings, the visual, motion, and aural cues, the response of the vehicle and its systems must all contribute.The disposition of the controls, switches and indicators must be geometrically correct,and their appearance and operation must correspond to their aircraft equivalents. Often, pilots are expected to wear the same clothing as worn in flight, and to follow the same operational procedures as used in flight.

All these factors contribute to providing a representative crew work-load. Clearly, in this class of training, subsidiary tasks or distractions can affect performance, and are included where appropriate.

A simulator which only represents a specific task can also fall into this category - for example, an air combat simulator. Basic skills, such as spatial orientation, manoeuvring relative to other aircraft for positive advantage, and a respect for ground proximity, can all be taught prior to flight training.However, to simulate multiple combat, it is usual to include additional players by using a simplified pilot console, with displays and controls sufficient foe the task, but without the full representation of combat seen by the pilot in a dome with all-round displays. Such a console is particularly good for an instructor who wishes to interact with the students in the domes. This configuration combines both categories of simulator. The pilots in the domes are being trained for air combat, and the pilots at the consoles are being taught air combat.

Realism in a flight simulator can be expensive. If a visual system is needed, it must have good resolution, high scene detail, texture, and a wide field of view. Time delays and low refresh rates are unacceptable, and so fast processing is essential. Aircraft-standard cockpits add to the cost. Even so, a high return on investment is usually obtained, both from development and training simulators.

5. Realism.

Realism is a quality that all manufacturers claim for their product. It is justifiable for the top of the range airline simulators,meeting FAA Levels C and D. Even though there is a considerable shortfall in the brightness and resolution of the best visual systems, compared to the real world, they are sufficiently good to justify describing the simulator as "realistic", when the combined effects of motion and aural cues, representative cockpit fit, and comprehensive dynamic modelling of the aircraft are taken into account.

Vendors of lower cost simulators often argue that the same level of transfer of training can be achieved with more simple solutions, in which the correspondence with the real world is not so good. By identifying the essential cues used by the pilot, and disregarding those which do not influence performance, a more cost-effective design is produced. It is further claimed that by exaggerating some features, such as colour contrast in a visual scene, other deficiencies can be masked. Stylised images may convey visual cues more efficiently than by trying to achieve realism, and failing to do so (ref 2). Using such distortions, more realistic pilot reactions are obtained.

Operators of research simulators expect more tolerance from their subject pilots than pilots under training might show. Their pilots are part of the research activity, interacting directly with the research scientists, and are expected to

define the standard of simulation needed to assess a particular task. In many cases, only those controls and displays which contribute to the task need be active. Even so, it is not unusual for the pilot of a new prototype to acknowledge the realism of the simulator - "I had been there before." At the same time, it must be said that very few R & D simulators qualify for Category B. Sadly, they are often operated with what are thought to be small deficiencies, such as no sound, or a badly set up visual system, because of external pressures.

The claim for realism even extends to devices aimed to amuse. Flight simulator games packages which run on personal computers offer such dramatic experiences as landing a Jumbo at Heathrow, or dogfighting in helicopters. The shortcomings of poor input and output devices, and slow update are conveniently forgotten. Even more incredible are the claims of realism made for pocket computers which simulate activities such as tennis and golf (even fishing!).

Realism is a subjective quality which defies measurement. Are there degrees of realism? There seems to be a point at which a near-step change in realism occurs. It takes the form of an unconscious change in mental state, in which an illusion is so compelling, that the subject changes his frame of reference - he (or she) is no longer aware of the illusion. Most people experience such a transition during a good film, perhaps with dramatic action, music, and a compelling narrative. The transition occurs less frequently in the theatre, and hardly ever occurs viewing TV. Entertainers such as conjurors hope to instil this state in the mind of their audience. It is sometimes called "suspense of disbelief".The range of susceptibility is wide - children are more likely to succumb to the effect. Pilots vary in their susceptibility, which probably explains some of the scatter seen in results depending on subjective assessments.

A flight simulator in Category B tries to create conditions suitable for suspense of disbelief. Motion platforms are driven in a manner to give cues most reminiscent of aircraft movements. The illusion of rotation and translation is complemented by the visual system, using vection to create the illusion. The effects of vection can be induced by visual cues alone, if the circumstances are right. One characteristic of vection is its step-like nature -once created, the illusion has to be broken (ref 3). For the purpose of this paper, a "realistic" simulator is one in which the step transition into suspended disbelief has been achieved.

6. Further Breakdown.

The new classification is seen on figure 2, and is further broken down on figures 3 and 4. Category A divides into the topics of Understanding, Learning, Investigating, and Optimising. Examples of devices/activities which fall into each category are given on figure 3. Their cerebral character is self-evident; successively, they become increasingly complex, and require a more sophisticated solution. Problems relating to mental workload fall into this Category.

Category B also divides into four topics: Co-ordination, Perception, Re-action, and Stress. They are fundamentally related to motor skills and conditioned reflexes. Again, progressive movement across the figure implies increasing complexity in the device to fulfil the training need. It is self-evident that workload associated with physical and emotional stress is in the domain of this class of simulator.

There is no reason why a new instructional device should not span both Categories. A military full mission simulator, or a civil simulator meeting FAA Level C or D, has the capability both to teach and to train the crew. The new breakdown simply

Figure 3

Category A

TEACHING - CEREBRAL SKILLS

UNDERSTANDING	LEARNING	INVESTIGATING	OPTIMISING
Principles	Features	Options	Operations
Books / Personal Computers	Cardboard flight decks	Interactive Devices / Computer Models	Network Simulation / Display Moding
Wall Charts	Animated Boards / Maintenance Trainers	System Rigs / Procedure Trainers	Air Traffic Simulation / Full Mission Simulation

Figure 4

Category B

TRAINING - MOTOR SKILLS

CO-ORDINATION	PERCEPTION	REACTION	STRESS
Control Actions	Cue Appreciation	Response to unexpected stimuli	Performance under pressure

IFR	Turbulence	Reduced Visibility	Illusions in Flight	Failures	Fatigue	Discomfort heat, smoke, g	Bad Weather

| Standard Manoeuvres | | Cockpit Displays and Moding | | Emergencies | | Conflicting Traffic | Return Fire |

Types of simulator

Centrifuge, Air Combat, Operational/Full Mission. Cabin Service/Emergency Exit, LOFT, Levels C,D

allows these activities to be related to the contribution of other devices, so that the division of work between them is an optimum.

7. Discussion.

The idea that a category A simulator should be accurate without being realistic is not contentious. It is less easy to accept that the category B simulator, which is realistic, need not be accurate. But the nature of an illusion is only to give the appearance of accuracy. Theatre is the habitat of illusion. The clothes and make-up that actors wear, their speech and movements are all exaggerated in relation to the real world. The stage itself is distorted; even when the action is subdued (a drawing-room comedy, perhaps) the scenery and props are not what they appear to be. Stage furniture is scaled up in size, the colours and lighting is chosen to obtain the maximum visual appeal.

Painters and artists use similar techniques. Landscapes often embody different scaling in the vertical and horizontal axes, and distance cues are emphasised by careful choice of colour. Photographs in monochrome rely on high levels of contrast, high resolution, and tricks of focus to be effective.

The specialists in computer graphics are also skilled in getting the most from the limited range of resolution, contrast, and brightness which is available to them. In real-time systems, they must also contend with the constraints imposed by the need for the fast refreshment of images. They share with painters the difficulty of creating the illusion of depth from a single plane display, although a they have the freedom to choose the distance of the image plane, by selection of the method of display. They can even consider whether or not binocular images are worthwhile.

In a similar manner, the engineers/physiologists responsible for re-creating motion stimuli in flight simulators contend with the inability of the motion systems to provide stimuli which are accurate in all respects (ref 4). The trade-offs between amplitude, frequency, performance and safety are well documented.

Although accuracy, in the sense of a 1:1 relationship with the world, can be sacrificed to achieve realism, a further constraint must be observed. There must be no conflict of cues, otherwise the illusion is destroyed. Unfortunately, the science of cueing is at a primitive stage, and some of the possible conflicts are not documented. Consequently, there is still an element of art in the design of a Category B simulator - an aspect which is not recognised by current Simulator Procurement practices.

8. Specific Examples.

The final acceptance of a simulator into Category B relies on subjective judgement. An astronaut was quoted recently as saying that none of the Shuttle simulations he had experienced came close to the real thing. Nothing could match the sense of power and drama when lying in the Shuttle, prior to and during lift-off. The build up of stress, the vibration and g forces was awe inspiring. Similarly, the sudden transition at engine shut off prior to orbit was equally dramatic- from 3g to 0g in less than a second, and from very high noise levels to total silence. No simulator could reproduce it.

Here is a challenge in simulator design - to excite emotions only encountered in very special circumstances- to reproduce conditions where mans' higher functions fail progressively. Vision tunnels, mental capacity reduces, secondary tasks are shed, and the sense of time distorts. The challenge can be met. There are simulators currently in use to train cabin crews for emergencies, for both fixed wing aircraft and helicopters, which are capable of providing harrowing experiences. Liberal use is made of noise, vibration, smoke, and water. They clearly qualify for category B.

a) <u>Simulator Sickness</u>. Simulator induced sickness has been a regular topic at conferences in the past few years. The definition of sickness requires that the simulator must produce in the occupants symptoms of nausea, dizziness, sweating, or worse, which would not be caused by similar exposure in flight (ref 5). A Steering Committee sponsored by NASA and the US Army is co-ordinating research on the topic. Although the US Air Force has not had trouble, the US Navy has reported many cases, particularly on helicopter and air combat simulators. Delayed onset of symptoms (flashback) have occasionally been noted, and some military operators restrict flying after exposure to certain types of simulator. Understanding the problem is hampered by its nature; it only occurs to a minority of pilots, on infrequent occasions. Consequently, field work requires impractically large numbers of trials to provide statistical evidence.

The A and B classification of simulators sheds light on this issue. Passive viewing of TV, and interactive devices for instruction or entertainment, do not pose medical problems of this nature. It could be inferred that all Category A simulators are immune. By definition, they are not sufficiently realistic to induce nausea. One mechanism to induce nausea is to present to the victim a conflict of information- as occurs for example in sea sickness. Category A simulators, and the associated teaching process, does not present information in this way.

Conversely, the criterion for Category B simulators is to achieve realism, to provide co-ordinated cues which correlate with flying, and to provide a stressful environment when necessary. It is understandable that in the most demanding cases, the state of technology of the simulator, or deficiencies in operation or maintenance, can cause problems. The irony is that the most ambitious simulators are the ones most likely to be affected.

b) <u>Networking</u>. Networking of simulators is now emerging as a practical way to extend personnel training for military applications (ref 6). The DARPA Simnet programme has proved it to be feasible. Ultimately, it is proposed that simulated aircraft, helicopters, tanks, and infantry will take part in exercises which are entirely inter-active. Already, the capability has been demonstrated to link several hundred simulators, separated by thousands of miles, to represent and study war scenarios. One of the drivers in achieving this goal is that the cost of equipment for each contestant must be low - a maximum of $0.1m has been set as a target.

One claim of this paper is that operator training, as distinct from operator teaching, is expensive, particularly if all modes are simulated. The above budget restriction severely limits the engineering standard of each element in the network. The visual system will be at the lower end of the market, and yet the requirements for operation on or close to the ground, and for many moving objects in the visual scene, are demanding. The physical conditions of battle are difficult to reproduce- apart from the discomfort and stress, when an element is disabled, the battle must continue. All considered the chance of a networked system qualifying for Category B is remote. (One element in isolation could qualify, if the cost per element rule is broken).

This shortfall detracts little from the value of the simulation. It opens up new territory in the area of Category A - learning, understanding, investigating, and optimising - and we have seen that these aspects benefit if the constraint of realism is removed. The message which emerges is that a networked system only provides part of the preparation for battle- the simulators for Category B training have yet to be designed.

The final element in the networked battle simulation is the ground soldier. If 'virtual reality' comes to fruition, this need can be met. Unfortunately, if the following discussion is correct, the chances are low.

c) <u>Virtual Reality</u>. By definition, Virtual Reality is a simulator technique which should be in Category B. In its present state, it has a long way to go to qualify. Apart from the 'surprise factor', the illusion of entering another world is not convincing, due in part to the low standards of current equipment (ref 7). Particularly distracting are the time delays associated with head position sensing and visual image update. The pointing code for translation in space, and the use of the virtual 'hand', take time to master. The advocates of the concept, in the main behavioural scientists and psychologists, admit these shortcomings, and attribute them to the low cost of the equipment demonstrated, and the current state of the art. If the technique is to progress beyond the realm of entertainment, computer games, and the psychadelic, into engineering applications, a fusion between the experience of the flight simulator community and the new wave virtual image proponents is essential.

Virtual Reality is potentially a minefield for all the symptoms of concern to the NASA Steering

Committee on Simulator Sicknes. It is likely that
they have not been encountered to date because the
current standard is poor, so that suspense of
reality, the criterion for Category B, has not
been achieved. But that is the goal, and
the minefield will have to be crossed before the
potential of virtual reality is realised.

9. Concluding Remarks.

The foregoing discussion has presented the basis
for an alternative classification of flight
simulators, using the distinction between cerebral
skills (mental activity) and motor skills
(co-ordination). It is suggested that for the
teaching of cerebral skills (Category A), accuracy
is important, and that for training in motor
skills (Category B), realism takes priority. To
say that accuracy is less important does not mean
that inaccuracy is admissible. It means that
simplifications are allowable if the pilot is
unaware that they have been made.

For example, fly-by-wire aircraft usually have
benign handling qualities throughout the flight
envelope; they would not be certified otherwise.
Even after failures, the pilot does not know what
equations are being solved in the flight control
computer - what he needs is advice on future
actions. Since the cost of providing aircraft
standard hardware and software is high, there is a
strong case to re-examine the merits of simulation
rather than stimulation.

Some of the technological issues in simulation
have been examined, in the light of this
classification. It would be interesting to extend
the examination to existing simulators or
facilities, to see how many can be accredited with
a Category B label. If there is a shortfall, the
improvements needed to qualify could be postulated.

A more important use concerns the specification of
new simulators or facilities. Avionics, applied to
flight control, systems management, and advanced
flight decks, has changed considerably the nature
of the crews' role; the crew complement, both
civil and military, has decreased in consequence.
More time is spent in a monitoring role, and less
in the control loop. Although the nature of the
task has changed, the crew work-load has not
significantly decreased at all times. The new
classification can assist in addressing the impact
of these changes (which are well recognised) on a
re-think of crew training needs.

10. References.

1. EM Boothe "Part Task Training Device
Standards"
Royal Aeronautical Society Conference on "Part
Task Trainers"   November 1990

2. FMP Working Group "Characteristics of Flight
Simulator Visual Systems"
AGARD AR 164   May 1981

3. LR Young "Visually Induced Motion in Flight
Simulation"
AGARD CP 249   October 1978

4. JB Sinacori "A Practical Approach to Motion
Simulation"
AIAA Paper 70-931   September 1973

5. RS Kennedy, JE Fowlkes "What does it mean when
we say that -Simulator Sickness is Polygenic and
Polysmptomatic?"
Image V Conference, Phoenix, Az.   June 1990

6. BW Thomas, CA Gainer "Simulation Networking: Low
Fidelity Simulation in US Army Aviation"
Royal Aeronautical Society Conference on "Progress
in Helicopter and VSTOL Aircraft Simulation"
                                  April/May   1990

7. P Elmer-Dewitt "(Mis)Adventures in Cyberspace"
Time Magazine   September 3, 1990
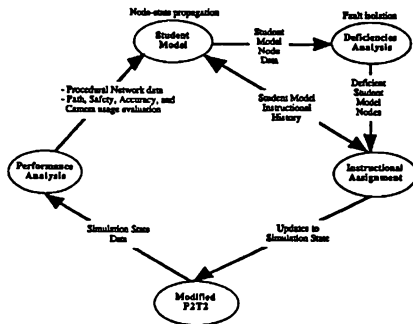
Figure 1: Functional Diagram of the P2T2 Intelligent Trainer

of the knowledges and skills are kept in the Student Model. Figure 2 shows the Domain Hierarchy.

Fault-isolation techniques are then used over the Domain Hierarchy to determine if we should test the student again at a lower point in the hierarchy, or if a probable weakness has been isolated and we should then tutor the student on that weakness. In testing, a "subscenario" is used that more specifically isolates an area of weakness, and again, student performance is monitored.

When such testing finally isolates a student weakness in the domain, a part task is used as remedial practice by the ITS. This task begins at a level of difficulty corresponding to the state of the student as determined by reasoning over the Student Model. Using these techniques, we expect to perform very efficient skill maintenance training.

Performance monitoring and evaluation: The P2T2 Intelligent Trainer monitors and evaluates student performance for efficiency, accuracy, safety, camera usage, and procedural correctness.
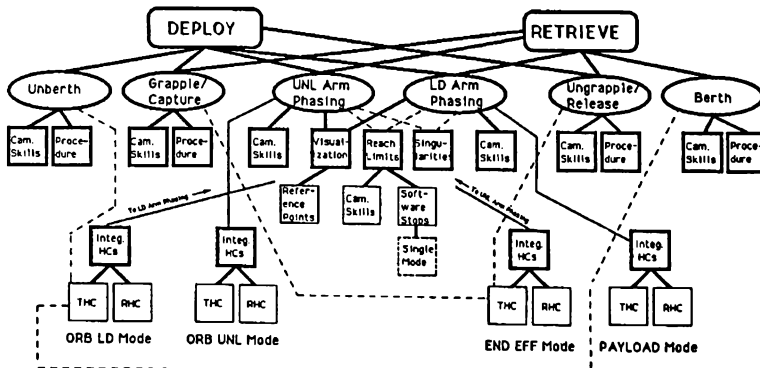
Figure 2: P2T2 Intelligent Trainer "Domain Hierarchy"

Innovation

We consider the following list to include the most important innovations of this work:

- Highly efficient skill maintenance is possible using the test-tutor paradigm, where more time-consuming tutoring is not performed until a specific problem is isolated through testing. In addition, using the fault-isolation techniques, even the testing is performed efficiently.

- Use of a series of subscenarios for testing to isolate the problem areas and then to tutor the specific student weakness is another innovative feature of this work.

- The P2T2 ITS integrates both an intelligent tutoring system and a part-task approach with an existing simulator.

- Tested instructional theories were used to develop the training approach, whereas the instructional component of most existing ITSs are based on a university instructor's idea of expert teaching. Even a psychology professor is not necessarily a good instructor!
    - Part task training research performed by Fredrickson and White (1989) was used for guidance on developing part tasks in training high performance skills
    - The Elaboration Theory, developed by Reigeluth and Merrill (1980), was used in ordering the part tasks for Tutorial Mode

- Diagnostics and tutoring are performed using the existing simulator. Very little modification of the P2T2 simulator was made in creating and integrating the trainer capabilities and functioning.

- Editable parameters for refining evaluation criteria and for supporting the training preferences of different NASA customers (shuttle, space station) are used. Thus, it may be determined through extensive usage that acceptable performance on a particular task takes three minutes instead of five, or that the student can fly the arm twice the shortest distance between two points and still be efficient. Non-programmers can change these values themselves.

- We use randomly-generated values (within certain ranges) and randomly-selected payloads to create large numbers of subscenarios and problems for students to solve, thus avoiding memorization of solutions.

- The P2T2 Intelligent Trainer maintains student data from one session to the next.

Summary Of Important Conclusions

The main conclusion of this research is that existing simulators can be modified to support very efficient training. In addition, the work can inform the design of new simulators, which rarely have built-in diagnostics, automatic scenario generation, or student models that guide the follow-on training. Efficiency is gained by automatically creating subscenarios to test the student, and not tutoring until a known weakness is determined. Students are then tutored using automatically-generated subscenarios as well.

Additional benefits can be found in off-loading from instructors the labor-intensive job of creating the subscenarios in the first place, or from standing by monitoring four CRTs while the students work with the simulator. In addition, the resulting evaluation is objective, consistent, and never forgets to mention a particular step left out of a procedure. Instructors are freed to work with students needing more direct help, or with more difficult problems (perhaps those involving complex explanations).

Future Research

- Need to automate knowledge acquisition so that an expert can enter domain knowledge directly

- When diagnosing the student, need to use statistics and a utility index (i.e., How worthwhile is it to remediate this problem?) to facilitate earlier and more accurate problem detection

- Need more research on how to structure domain knowledge (e.g., hierarchy or network?)

- Could use training objectives to further refine selection of subscenarios.

- Transfer the ITS to a different simulator (e.g., a simulator of the Space Station arm, or the Shuttle Mission Simulator)

Acknowledgements

References

Frederiksen, John R. and White, Barbara Y., "An approach to training based upon principled task decomposition," *Acta Psychologica*, 71 (1989), pp. 89-146.

Huggins, A.W.F., et al. "Intelligent Technology for ATC Training," In *Symposium on Air Traffic Control Training for Tomorrow's Technology*, December 6-7, 1988, Oklahoma City, OK, pp. 128-147.

Regian, J. Wesley, Dennis, Margaret, and Shute, Valerie J., "INFLITE: An Intelligent Instrument Flight Trainer with Computer Generated Speech," *Proceedings 1989 Conference on Technology and Innovations in Training and Education*, March 6-9, 1989, Atlanta, GA, pp. 407-414.

Reigeluth, C.M., & Stein, F.S., "The Elaboration Theory of Instruction." In C.M. Reigeluth (Ed.), *Instructional-Design Theories and Models: An Overview of their Current Status*. Hillsdale, N.J.: Lawrence Erlbaum Associates, 1983.

Reigeluth, C.M., & Rodgers, C.A., "The Elaboration Theory of Instruction: Prescriptions for Task Analysis and Design," *NSPI Journal*, February, 1980, pp. 16-25.

ENGINEERING TRAINING PROPOSAL

Katherine M. Constable

Employee Training and Development
Boeing Commercial Airplane Group
Seattle, Washington

ABSTRACT

The need to train a more diverse workforce in simulation concepts and techniques has been growing with the increasing use of simulation in the design, development, and testing of aircraft and their components.

Existing academic training, in most instances, does not meet current simulation training needs. An inquiry was conducted at Boeing to determine if a basic core of courses could be taught in a classroom if hands-on experience is provided.

The prevalent method training in the industry is unstructured on-the-job training. This must be replaced by structured on-the-job training developed and administered by the simulation department.

PREFACE

The problem

There is no industry-wide standard method of training simulation engineers. Indeed, most training for this field is unstructured on-the-job training. This is the principal technique used today to train all personnel in all aspects of the simulation milieu. Some technicians have been trained by the military or by technical schools, with an average time investment of about two years. Most people, however, learn their trade by following someone around, watching, and asking questions.

Unstructured on-the-job training is neither speedy nor comprehensive and it is **not** cost-effective. There is no organized approach to introducing simulation concepts. The existing method of training simulation engineers and engineers who utilize the simulations is costly, time-consuming, and not highly effective. It is in the interest of continuous quality improvement and, in fact, survival in this highly competitive time, that a more cost-effective and thorough method of training engineers and technicians to work in the simulation environment is required.

BACKGROUND

Definition

What is simulation? In its strictest sense, it is just a model of a system. It can be anything from a mathematical model of a nuclear blast to a simulation of an aircraft. An aircraft simulation can take the form of a full scale, six degrees-of-motion, full flight simulator, equipped with an eight million dollar visual system that is analogous to the airplane in all phases of flight, to a simple test bench connected to a small computer.

The new Boeing airplane, the 777, will have more engineers involved in more kinds of simulation than any other aircraft in Boeing history. Simulation is becoming more important in the future development of aircraft. Experienced engineers with no previous background in simulation are working with simulations for the first time, and are hampered by a lack of training.

Cost, quality, and preparation

Why use simulation? Cost savings, quality improvement, improved accuracy of predicted data, and a greater chance of meeting schedules are goals more easily met by using simulation. Before the 777 ever lifts off the runway, the pilots will have flown a 777 simulator for hundreds of hours. They will fly in a simulated cockpit "cab" which will be a replica of the cockpit of the airplane they will eventually fly. The handling characteristics of the simulated airplane will be the best engineering solution possible given the wind-tunnel data, the mathematical calculations, and mechanical engineering expertise. The pilots will have a chance to become familiar with every system on the flight deck; where they are located, and how they work.

In addition to pilot checking and training in the simulated cabs, there will be hundreds of thousands of hours of "off-line" (as opposed to cab) simulation time in laboratories and on desk-tops with new simulation tools.

Safety

When the actual airplane takes off, every system in it will have been tested on a lab test bench which has been connected to a computer simulation or in a simulator. The simulation will enable the system to respond as though it were actually flying in an airplane. The embedded avionics will be tested in a broad spectrum of flight profiles and environmental conditions before the system is considered safe to be installed in the airplane. The test engineers can verify the software and make corrections more easily in the laboratory environment. This testing would be difficult or even dangerous to accomplish if the system were installed in an airplane.

Evolution

Instruments which were formerly self-contained and hydraulically, electronically, or mechanically driven, have been replaced by digital displays coupled by digital data links, such as ARINC 429 or ARINC 629 (standard bus protocols), to other systems. The embedded software is hosted on dedicated airborne computers called LRU's (line replaceable units).

In the past, before digital avionics and simulators, systems were installed in the airplane and much time was spent developing the system during flight test. When Boeing began to install digital avionics in airplanes, starting with the 757 and 767, they were aware that a new approach to testing systems must be attempted. The change to digital avionics was of far greater complexity than the change from reciprocating engines to jet engines.

A solution

Elaborate laboratories were developed and sophisticated test benches were designed. A system was then connected to a computer in such a way that the system could respond as though it were being driven in the aircraft environment. The technology developed in the early days of laboratory testing has continued to grow and be refined as avionics systems and computers became more sophisticated.

Considerations

What thinking took place that allowed engineers to accept the validity of such testing? What kind of people developed the mathematical model of the airplane? Who taught them how to do it? Who taught the technician how to design the work station interface? Who maintained all the hardware and software, and how did the people learn all these skills?

Academic training

What academic training is required to become a simulation engineer? The engineers and mathematicians who developed the models which are incorporated into a computer come from all disciplines. Aeronautical engineers define the aeronautical part of the model. Electrical engineers have the expertise to define electrical systems and servo-mechanisms. Mathematicians assist in developing equations of motion and equations used in computing the position of the "aircraft" on or above the earth. Engineers with degrees in control theory assist in the development of algorithms which are used in a LRU (such as an autopilot or a flight control computer) or in the simulation of an LRU. Mechanical engineers create algorithms defining the engines. Software engineers assist in coding models and in designing computers' executive operating systems, that software which enables all the pieces to play together.

These are traditional engineering fields and most schools provide adequate preparation for the engineer to assume responsibility for developing his/her piece of the simulation. Few schools, however, prepare the engineer to work in the simulation environment and few engineers understand the basic concepts of simulation. The aggregate team is often, in some respects, like the seven blind men evaluating the elephant. To the mathematician, the simulator is a rather useful collection of equations; to the software engineer it is a bunch of subroutines.

Most simulation facilities, including those outside of Boeing, do not have structured training programs for anyone involved in the simulation environment. Nor are there any plans in place to provide training other than on-the-job. This lack of structured training will become increasingly more expensive and is counter to the ideals of quality and productivity. Depending on the company and the job, most individuals surveyed gave estimates of six months to five years to train a competent, productive simulation engineer.

Inquiry

How can a cost-effective, general purpose training program which would be of use across such a melange of disciplines and skills, be developed for people with such diverse needs? An inquiry was developed and sent to twenty groups within Boeing to determine the interest in a basic simulation concepts course. Seventeen of the twenty inquiries were returned. The premise was that the requirements would be too diverse and the numbers too small to make it practical to develop a comprehensive course which would

cover enough topics to be of use to the majority. This premise appears to have been only partially correct.

Topics

The inquiry posed questions about the topics to be covered, the previous method of training engineers in the simulation environment, the length of time required for adequate training, and the extent to which the group was involved with simulation. The inquiries were sent to groups which designed and developed simulations and to groups which used simulation. As Boeing is a large company and has a number of types of simulation being utilized, it seemed advisable to query commercial as well as military simulation facilities. The end users responding to the inquiry were a mixture of experienced and novice simulation users, with one of the groups responding as a future but non current user of simulation. The simulation topics presented for selection were the following:

- Basic Aerodynamics
- Concepts of Modeling
- Basics of Visual Systems
- Flight Controls
- Equations of Motion
- Basics of Control Systems
- Basics of Motion Systems

The following chart indicates the responses received.

Requirement for the topic
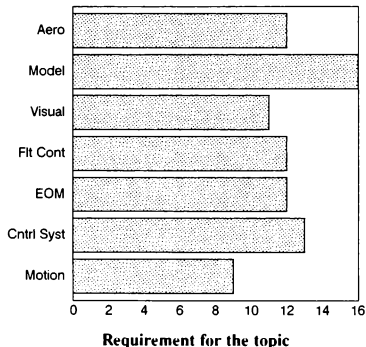
There was also space provided on the form for additional suggested topics. Among the most obvious omissions from the inquiry were the following topics:

- Background, History, and Terminology
- Numerical Methods
- Troubleshooting
- Data Collection and Reduction
- Verification and Validation
- Certification

The History, Background, Terminology, and Numerical Methods topics above would lend themselves readily to a generic simulation course. It would be useful to introduce the kinds of simulation while presenting background information. The remainder of the topics are generally more specific to a particular environment and would be more difficult to approach for that reason.

Current method of training

The existing method of training for most of the organizations responding was on-the-job, with 75% of employees receiving this type of training. One group has not yet been involved in simulation and is anticipating needing some sort of introduction to the basic concepts. A small percentage of the respondents stated that some of their people were trained outside of Boeing and that the training process involved both on-the-job and classroom training.

Time required for training

There was no clear consensus on the length of time required to fully train a simulation engineer. While the sample is small and restricted to the Boeing Company, it appears that the majority of managers indicate that 6 to 12 months is sufficient.
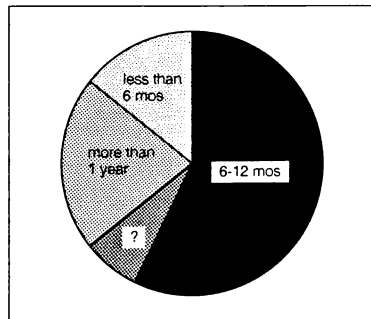
Time to train

It was interesting to note that some of the more experienced simulation managers felt that it took longer

to train a new simulation engineer than some of the less experienced managers.

As could be expected, groups new to the simulation environment had no idea as to the amount of time required to train an employee new to the field.

Number of individuals requiring training

The response to this question was quite interesting. Users of simulation felt the need for more training than those involved in the development or maintenance of simulation. All respondents indicated that all employees new to the simulation environment would benefit from taking an introductory class, while users of simulation felt that all employees, regardless of their experience, would benefit from a "big picture."

Number to be Trained
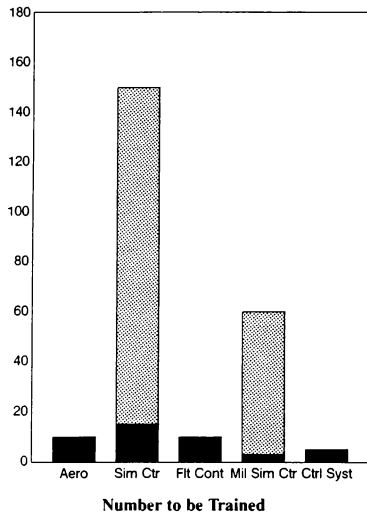
As can be seen from the above bar chart, the amount of training to be done is not a significant percentage of the engineering population. (The number to be trained is shown in black.)

SOLUTION TO TRAINING PROBLEM

Structured on-the-job training

This approach has been used successfully by a number of companies and has become an accepted approach

to teaching highly technical materials. Both mentoring and structured on-the-job training require the assistance of the most highly qualified technical people in a group. These are normally the busiest people and the ones that a manager is least willing to make available to assist with training. The operative word is "assist" as not all technical experts are willing and/or able to teach. They must be available for course development, consulting, and evaluation of courses.

In order to be effective, structured on-the-job training must be:

- in support of company goals.
- defined by the simulation department.
- developed by the simulation department with the assistance of training professionals.
- in support of a clearly defined career path.
- taught by a topic expert (and not by training or human resource personnel).
- in support of company quality policies.

First-line managers will become coaches as well as providing technical support. This means that they will have to be selected for their coaching abilities or be trained to coach.

Studies indicate that a student retains 10 percent of what he hears and 90 percent of what he "hears and does." If simulation topics are to be taught in a classroom, then some means must be devised for "hands-on" exercises. At Boeing there are several in-house tools which can be used to support the learning of such things as concepts of modeling and control theory. One of the tools allows the engineer to create a simple control system graphically and then to perform several types of analysis on the system. Another tool, also graphically based, allows the engineer to create an entire simulation by block diagrams which can then be automatically translated into either FORTRAN or Ada code. This code can then be incorporated into a simulation which can drive a test bench, a simulation cab, hardware-in-the-loop, or run as a stand-alone model.

Technical experts

Individuals who are "experts" in their own arenas would assist in determining what skills and knowledge are necessary in order to do the job effectively. Meanwhile, the training department could establish a program of structured mentoring — coaching simulation experts in the skills involved in effective mentoring. Initial skills testing and methods of measuring success must be established.

When a new engineer comes on board, a general skills evaluation test could be administered to determine what areas of knowledge have already been mastered and what areas of learning must be enhanced. This would prevent unnecessary time spent covering topics that the engineer has already mastered. A career and training plan must be developed taking in to consideration the individual's general interest, ability, and career goals, as well as the needs of the organization.

Mentoring

After a skills evaluation, the individual could be paired with the first of several mentors. The mentor would be a specially trained expert in one of the disciplines in the simulation field. The new engineer would then be trained by a combination of special classes and coaching from his mentor. The mentor would be trained in giving feedback to the new employee and in providing the employee's supervisor with regular information on the employee's progress in the training plan.

When both the mentor and the manager felt that the employee was ready for a new discipline, the employee would be assigned a new coach. The original mentor would continue to guide the new employee until the training process was finished but would only act as coach in his/her own area of expertise. A mentor could have more than one engineer as a mentee.

Classroom training could involve general simulation topics such as: "What are the equations of motion?" "What is real-time?" "What is transport delay?" A combination of classroom instruction (or perhaps individual training modules), accompanied by mentoring and coaching should assist in providing the new employee with a comprehensive, measurable, and more rapid means of being assimilated into the simulation environment.

Expected benefits

Expected benefits of structured on-the-job training would include the following:

- Just-in-time training
- Better use of skills for new and existing employees
- Less time involved in bringing a new employee up to acceptable performance
- A means of passing on critical skills and knowledge held by more experienced (and older) employees so that knowledge is not lost when employees retire or leave
- A means of developing promising employees into top performers
- Greater job satisfaction

These benefits could assist in keeping promising employees who feel that their current job offers no future.

Conclusion

Training in the simulation environment is the responsibility of the simulation department. The majority of simulation training has been unstructured on-the-job, which is the most costly and least effective method of training.

It is necessary to examine the performance, skills, and attributes of top performance in order to determine areas where training is needed. Career paths, and the training to support those paths, must be clearly defined. Training must be supported by all levels of management and must not be considered as an add-on.

The development of curriculum and course content is the responsibility of the training department. However, training professionals should be consulted in the actual development of course materials. A basic core of concepts of simulation classes could be developed but must contain opportunities for hands-on experience.

Continuous quality improvement requires that better, more cost effective training be developed. There is no single curriculum which would meet all needs.

A COMPUTER GENERATED HELICOPTER FOR AIR TO AIR COMBAT

Amnon Katz, Ph.D.*, Brett E. Butler+ and Daniel E. M. Allen+
McDonnell Douglas Helicopter Company
Mesa, Arizona 85205-9797

ABSTRACT

A prototype of Intelligent Player, a computer generated simulated helicopter for air to air combat, has been implemented in real time, running against a high fidelity manned helicopter simulator. This paper addresses implementation issues. Certain aspects of the control logic (override logic) are also covered.

I INTRODUCTION

Intelligent Player is a computer generated Helicopter intended for the purpose of populating and enriching the tactical environment of a high fidelity manned simulator. Intelligent Player's main function (and the only one implemented so far) is to engage the manned simulator in one on one air combat. The logic engine driving Intelligent Player is based on chess type lookahead adapted to the air combat application. This logic engine has been described elsewhere [1]. It differs from other similar applications [2,3,4] mainly in:

1. The decisions in the lookup tree are made by the two adversaries sequentially rather than simulataneously.
2. The scoring is strictly within the framework of the theory of probabilty.

Reference 1 reported also some results obtained off line by running two players against each other. But at that time, Intelligent Player had not been run in real time interactively with a manned simulator. This has since been accomplished in the Advanced Apache Copilot Gunner simulator at the Mesa facility of McDonnell Douglas Helicopter

---

* Presently Professor of Aerospace Engineering, The University of Alabama, Tuscaloosa, AL 35487-0280. Member AIAA.
+ Member Technical Staff.

Company. The purpose of this paper is to report on the prototype implementation, to address certain issues of logic not covered in reference 1, and to assess the capabilities, limitations, and future prospects of Intelligent Player.

II THE REAL TIME INTERFACE

The MDHC simulation facility is a distributed and networked system (Figure 1). The standard building blocks are VME chassis, each containing multiple 68020 microprocessor boards. The many different chassis are networked by a Proteon Pronet 80 token ring local area network for real time communications. They are also connected by an Ethernet.

One of the system building blocks is the Tactical Mission Computer System (TMCS). This particular chassis supports one or more simulators by providing the tactical environment. The TMCS maintains a data base of active players, determines which players have clear line of sight to which other players, and computes the weapons flyouts for all the players it serves.

The TMCS generates some players within itself. Other players are computed outside the TMCS. These include full fidelity dome simulators and limited fidelity desktop stations. The TMCS communicates with external players by Pronet. Each player puts its current information packet on the token ring. The TMCS absorbs this information into its data base. It then broadcasts a packet containing information about all players that all players receive. The TMCS does not know or care what an external player physically is or where it is located. The Intelligent player is presented to the TMCS as just one other external player.

The problem of presenting an alien and intrinsically asynchronous external player had arisen once before in the context of long haul networking [5,6,7]. The remote player information

had to be put on the local net in the format and at the frequency that the local standard requires. This was accomplished by a dedicated VME chassis called the Gateway (GW).

The GW is designed to appear as one or more additional external players to the TMCS. This is accomplished by appropriate Pronet communications. However, the prototyping of the GW was done in conjunction with the the Advanced Apache Copilot Gunner simulator, which at the time had not been upgraded to use pronet. It employed a Gould 9780 rather than VME chassis for real time host. The TMCS communicated with the Gould through an HSD link. For this reason the Prototype GW communicated with the TMCS by Ethernet. This was done by the MDHC Stealth Protocol [8,9]. Using the Stealth Protocol, the GW "peeks" and "pokes" TMCS memory. The GW reads the TMCS players data base and writes into it transparently to the TMCS.

The prototyping of the Intelligent Player used the same Advanced Apache Copilot Gunner station for the same reason, namely that it was a stable simulator not undergoing development at the time. The prototype IP used the actual prototype GW.

The GW is a VME chassis and uses three processor boards:

1. The GATEWAY board, which mediates between the longhaul and local nets.
2. The COMMUNICATIONS board that hosts the modem driver and is used to set up communications with the TMCS.
3. The COMMAND & CONTROL (C&C) board that supports the Ethernet functions. these include downloading the whole chassis as well as hosting the Stealth master. The last board is a standard part of all MDHC chassis.

The Gateway board operates synchronously and
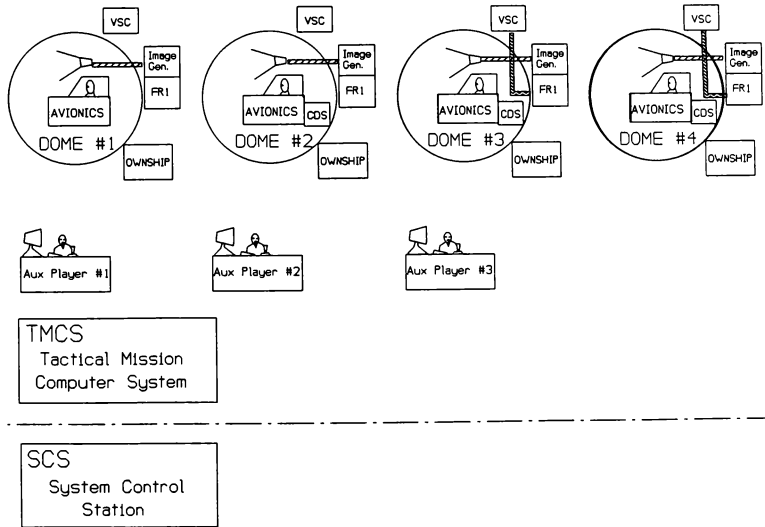
Figure 1: The McDonnell Douglas Helicopter Company Distributed Simulation Facility.

communicates with the TMCS at 60Hz. The communications and C&C boards provide on-demand services. In the IP application a fourth board was added:

4. The LOOKAHEAD board that runs continuously and asynchronously and produces maneuver decisions roughly at 0.6Hz.

The function of the gateway board remains to mediate bewtween the 60Hz TMCS and the asynchronous LOOKAHEAD board. Specifically the functions of the Gateway board in the IP application are:

1. Read ownship coordinates every frame and make them available to the *Lookahead* and *Override* logic.
2. Read the maneuver decision last produced by the LOOKAHEAD board.
3. Process the maneuver decision through the *Override* logic.
4. Run the *Maneuver Interpreter*, which translates the maneuver chosen into frame by frame control

inputs for the flight model.
5. Run the IP flight model and supply the TMCS with an updated state every frame.

The flight model is discussed elsewhere [10]. The *Maneuver Interpreter* will be the subject of a future paper. The *Override* logic is addressed in the next section.

The function of the Command and Control board is unchanged. The Communications board is relieved of modem communications. In the final IP it acquires instead the task of Pronet communications. In the prototype it was lightly loaded.

In the prototype implementation only the LOOKAHEAD board used its full computational throughput. Nevertheless the four board architecture described here is very effective for prototyping and low volume production, where software development and integration makes up the bulk of the cost. The functions of the different boards are parallel and independent, making development and debugging easy, and enhancing system reliability.
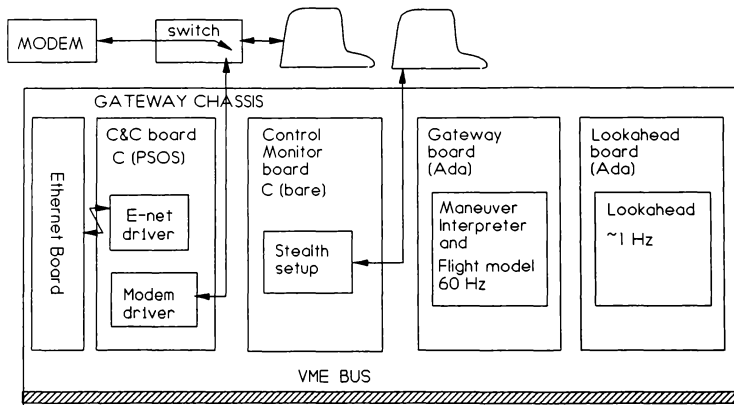
INTELLIGENT PLAYER CONFIGURATION

ipsw

Figure 2: The Gateway Computer.

III OVERRIDE LOGIC

The Lookahed logic we employ [1] selects among 11 extreme maneuvers. The resulting bang-bang style of control is typical of dogfighting as well as usual for many problems of optimal control. Still, there are circumstances under which this logic must be supplemented or modified for a reasonable IP to emerge. The implementation described here recognized three circumstances calling for overriding the lookahead decision:

1. The AIM maneuver. Combat tactics will vary with the weapons used. For simplicity, this implementation employed a gun fixed to the fuselage. The purpose of the maneuvers selected by the Lookahead was to provide a fire opportunity. However, the bang-bang style of control cannot produce an aim accurate enough to hit a target, let alone maintain that aim long enough to inflict damage. For this purpose a specific AIM maneuver was introduced, which takes account of the adversary's position and employs proportional navigation to bring the gun to bear. The AIM maneuver was included in the repertory of the real-time MANEUVER INTERPRETER on the Gateway board (but not the one used by LOOKAHEAD). Whenever LOOKAHEAD produces an aim within a certain tolerance, any further LOOKAHEAD decisions are overridden with the AIM maneuver. The purpose of the AIM maneuver is to improve the aim. When a different, much tighter, fire tolerance is achieved, a fire command is issued. The TMCS then computes and flies the projectiles, which are displayed to the pilot of the manned simulator. In case of a hit a fireball is also shown. The override remains in effect so long as the aim tolerance is maintained. Once it is lost, the maneuver commanded by LOOKAHEAD (which has never stopped functioning) is again honored.

2. COLLISION AVOIDANCE. The lookahead logic does not avoid mid air collisions. For simplicity, the effect of these is not currently implemented in the lookahead. In any case, based as it is on zero sum game theory [11], LOOKAHEAD would regard a collision as a draw, as acceptable as no blood. In reality, adversary pilots cooperate in avoiding collisions. This function is provided for the IP by the COLLISION AVOIDANCE maneuver, again available to the real-time MANEUVER INTERPRETER, but not to its Lookahead counterpart. Whenever a collision is imminent, the maneuver selected by LOOKAHEAD is overriden and COLLISION AVOIDANCE substituted. When the threat of collision is no longer a factor, the LOOKAHEAD selection is again honored.

3. A Maneuver Outlives its Usefulness. It has been discussed in reference 1 that maneuvers, such as turns, must not be held too long. There it was suggested that in the lookahead, different branches need differing lengths. This, however, does not help in real time. The turn previously selected may now be detrimental, but LOOKAHEAD, due to its computational load, is unable to produce a new selection yet. This problem was solved heuristically, by overriding maneuvers that are overdue for revision with straight and level flight.

Figure 3 shows the flow of OVERRIDE logic.
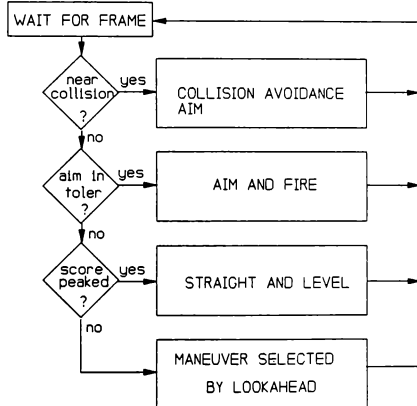
Figure 3: Override Logic.

IV GENERAL REMARKS

Intelligent Player has flown repeatedly in the Advanced Apache Copilot Gunner dome. It

persisted in bearing on the ownship and strafing it when it could obtain the required aim.

IP has not undergone any systematic evaluation of combat tactics against experienced pilots. It may be too early to attempt this. IP's tactics are elementary, and resemble a crude style of WWI circling dogfight. IP was at a disadvantage in that it was restricted to maintain a minimum airspeed. This limitation was imposed due to lack of adequate development of the MANEUVER INTERPRETER to cover hover. This point seemed to be IP's achilles' heel, since it restricts his turn rate. The ownship, by slowing to a hover and turning in place, could outturn the IP and score more hits than it suffered.

The biggest limitation of the prototype IP was the depth of its lookahead search, which was limited to a single ply and produced a decision every 0.6 seconds. One ply means that IP evaluated the situation arising from flying any of his eleven manevers against his opponent's current maneuver. In a two ply situation the IP would evaluate each of his 11 choices against the eleven responses by the opponent a half second later. Design studies showed that a two ply lookahead producing maneuver decisions once a second is within reach. However, it is the three ply lookahead that is the real breakthrough. Here the IP can evaluate a combination of two decisions by himself against one by his opponent. Two successive decisions are the minimum required to formulate a plan: "I'll do this now, so I can get him later..." Off line studies have shown that a three ply Lookahead enjoys an overwhelming advantage over a one ply lookahead.

Between the offline studies and the real time prototype, it appears that all the elements are there for a useful and effective Intelligent Player. Continued development and integration can put it together.

REFERENCES

1. Katz, A., and Ross, A.: "One on one Helicopter Combat Simulated by Chess Type Lookahead", AIAA Flight Simulation Technologies, Boston, Aug 1989, p 357. to be published in the Journal of Aircraft, 28, no. 2, p158 (1991).

2. Falco, M., Hinz, H., and Sobierajski, F. "Combat Performance for Light Helicopter Concepts", Grumman Report RE-616 (December 1980).

3. Austin, F., Carbone, G., Falco, M., and Hinz, H. "Automated Maneuvering Decisions for Air to Air Combat", Grumman Report RE-742 (November 1987).

4. Austin, F., Carbone, G., Falco, M., Hinz, H., and Lewis, M. "Game Theory for Automated Maneuvering During Air-to-Air Combat", Journal of Guidance, Control, and Dynamics, Vol 13 #6, pp 1143-1149, Nov-Dec 1990.

5. Katz, A.,, D. Allen, and N. Rao "Networking in a Distributed Simulation System", Proceedings of the 11th Industry/Interservice Training Conference, Ft. Worth, Nov 1989, p 70.

6. Katz, A., and W. E. Richeson "Simulation Networking", The Fighter Helicopter Conference, London, Jan 1990.

7. Katz, A., "Simulation Networking", AIAA Simulation Technologies Confference, Dayton, Sep 1990, p 123.

8. Allen, D. M., and Rao, N., "Real Time Data Collection and Control in a Distributed Simulator System using Ethernet TCP/IP", SAE Aerotech89, paper #892356, September 1989.

9. Katz, A., Allen, D. M., and Rao, N., "Networking in a Distributed Simulation System", Proceedings of the 11th Industry/Interservice Training Conference, Ft. Worth, Nov 1989, p 70.

10. A. Katz and B. Butler "A Flight Model for Unmanned Simulated Helicopters", 1990, to be published in the Journal of Aircraft.

11. Von Neumann, J., and Morgenstern, O., *Theory of Games and Economic Behavior* Princeton University Press, Princeton, N.J. 1944.

MULTI-AIRCRAFT SIMULATION FOR THE TRAINING OF
FLIGHT AIRCREW AND AIR TRAFFIC CONTROL PERSONNEL

G. D. Gibb, J. W. Blanchard, and A. J. Kornecki
Airway Science Simulation Laboratory
Embry-Riddle Aeronautical University
Daytona Beach, Florida

ABSTRACT
Traditionally, flight personnel and air traffic controllers are trained independently of one another and interact only in the live operational environment. However, computer architecture and networking have advanced sufficiently in recent years to allow for development of integrated systems in a simulated environment. This paper discusses the networking of flight simulators and air traffic control (ATC) stations to emulate the major components of an airspace environment. Flight operations, air traffic functions, and meteorological factors are accurately integrated to form the basis of the simulation system. By engineering air traffic control stations and networking software to commercially available meteorological computers and flight simulators, a three-dimensional real-time airspace was created. The engineering design specifications, associated hardware, and technological problems experienced will be addressed.

INTRODUCTION
The training of air traffic control and flight crew personnel in a simulated environment is generally conducted separately for each specialization. Both the military and civilian (Federal Aviation Administration (FAA) Academy in Oklahoma City) air traffic control environments generate "traffic" in their simulations by using pseudo-pilots, or computer-generated inputs. The vast majority of pilot simulation training is directed toward instrument proficiency and rarely considers the effect of other traffic or air traffic control communications and instructions. Typically, a flight instructor or performance evaluator will assume the roles of air traffic control positions. This type of training is common in both military initial pilot training programs and in the civilian sector. Nevertheless, the simulated flight

occurs in a vacuum devoid of the critical elements present in the actual operational environment.

The purpose of developing a multi-aircraft simulation was not to increase the level of fidelity of aviation training, but to provide a simulated environment that permitted forms of training otherwise not available or feasible in an operational setting. Clearly, extraordinary levels of fidelity are available in commercial motion-based flight simulators with visual systems for pilots. Air traffic controller trainers have been witness to the recent installation of a control tower cab simulator at the FAA Academy that features variable weather and time of day conditions with voice activated aircraft. These systems do not involve the interaction of key personnel in the airspace environment yet numerous mishaps can be traced to ineffective pilot-controller interactions as a prominent cause. The Avianca B-707 incident in New York in 1989 is a case in point. Although exhaustion of fuel was the physical reason for the crash, inadequate and ineffective air-ground communication was an underlying cause. The simulation system described in the following sections outlines how an airspace was constructed that allows the interaction of controllers and pilots. The design of the system was engineered such that aircraft (flight simulators) share the same airspace. Within the simulation system a flight simulator is no longer a stand-alone training device but an integral component of the system that impacts the operation of other simulators and is impacted respectfully by their operations. Air traffic control simulators are the nucleus of the system and provide the visual cues of the airspace and all operating aircraft. Data input from the flight simulators to the air traffic control displays that depict the location of simulators provide the foundation for the simulated airspace.

The primary advantage of this integrated simulation system is the opportunity to offer training in atypical situations. Airports may be closed, active runways can be changed or closed,

aircraft may experience on-board emergencies, unanticipated targets may enter arrival/departure/approach areas, or meteorological conditions can vary. In each instance, a training environment is established that requires the effective communication and coordination of several participants. These opportunities do not exist with stand-alone devices, nor is it feasible to conduct them in the operational environment. Secondary uses of the multi-aircraft simulation include: 1) expansion of cockpit resource management training for flight crew by inclusion of air traffic control communications, 2) investigation of human factors issues by the capability to control, standardize, and measure numerous variables, as well as the ability to replicate conditions over several experimental trials, and 3) enhancing the initial training of aviation personnel.

INTRODUCTION TO ATCSS DESIGN

The Air Traffic Control Simulation System (ATCSS) was designed under the concept of communicating over an EtherNet or other large scale distributed communications network. One of the important features of this system is its real-time nature and transfer of data. This includes pilots and air traffic controllers, as well as the other facilities in the airspace system communicating over voice channels. To accommodate voice, an eight bit word length was chosen as a standard for conversation. Digital format conversion is used to transfer through the network. This will be discussed in the System Design section, but was a very important consideration in system design.

The inclusion of part-task trainers for flight operations was necessary, therefore, a large portion of the system design addressed how they would be used. A third facet of the system design that needed to be considered was the inclusion of real-time weather into the simulation. This weather data could be updated automatically through the use of a meteorological facility networked to the system or could be updated manually by the insertion of pre-recorded weather phenomenon. This will also be discussed in more detail in the System Design section.

SYSTEM DESIGN
Flight Simulators

The flight simulators used in this test bed of the ATCSS were Frasca flight simulators and a 1970's vintage Boeing 707 cockpit procedures trainer made by Singer-Link. All of these

simulators operate with an analog or TTL output. It was necessary to devise a standard conversion package which a host computer could use to poll these simulators for the required information. Currently, the system is hosted by a MicroVax II operating on a serial interface to each of the systems (STAR Architecture). This is an important feature of the flight simulator section because it allows the simulators to be polled as required by ATCSS (rather than consistent sending), thereby reducing the load on the Central Processing Unit (CPU) of the flight simulator (and producing a "Mode S" type data acquisition system). A major requirement of the system was that the polling of the simulator be transparent to the pilot.

Part-task Trainers

The part-task trainers that are in use by the Airway Science Simulation Laboratory (ASSL) at this time consist of the Novel Twist Pilot Station attached to an IBM PS-2 computer running software specifically designed by the Laboratory for the practice of instrument approaches. These part-task simulators allow the pilots to participate in the simulation and receive radar and weather services as well as communicate with other pilots; however, they do not require the large expense normally found in a cockpit procedures trainer or flight simulator. This is an important feature when discussing the concept of low cost flight training devices for use by pilots in the initial pilot training sector, and specifically, the instrument rating.

ATC Simulators

The ATC simulators used in the simulation are simulators which have certain design features similar to the Initial Sector Suite System (ISSS) or the Automated En Route Air Traffic Control (AERA) II system. Both of these systems include features which have been evaluated for use in generic applications to the ATC environment. Some of the more notable features which are included in the ATCSS are: the ability to hand zoom, the ability to model objects three dimensionally, the ability to forecast or indicate predicted routes of aircraft including velocity vector information, the use of color coupled with symbology, and the use of flight strip target matching and electronic flight strips. Other features of this system include the distributed node environment operating on an EtherNet backbone so that data is made available to those who require it, rather than using an internet address or direct

serial communications line. This enhances the speed of the system and allows for sectors to be moved at random by the individual operator at the workstation. A limitation to this design is system capacity (10 mbps). The workstations utilized in the ATCSS are the IRIS 4D-20 and 4D-25G versions.

Weather
The weather facility used in conjunction with the ATCSS includes the same facilities that are available to Flight Service Stations (FSS) in the National Airspace System (NAS) structure. Equipment is manufactured by Kavouras, Inc. and data is received on a downlink through traditional satellite communications (WESTAR IV). This weather data can then be infused into the simulation to provide limited portions of the information to the air traffic controllers. The Flight Service Station acts as a recipient for flight strips when pilots desire to participate in the system. In doing so, they would call the ATCSS Flight Service Station and file a flight plan as they traditionally would in an operational environment. This is how the information is entered into the system initially. This design feature was very important because it allowed students participating in the ATCSS-FSS weather facility the opportunity to perform the roles traditionally performed by Flight Service Stations in the National Airspace System. Additionally, it added a level of fidelity to the system not currently available in a traditional pilot-controller training environment.

Central Processing
The MicroVax II is connected to the EtherNet and therefore provides data collected from the flight simulators over the serial STAR network to the air traffic control stations as required. This occurs in the form of a broadcast message. Our central processor, the MicroVax II, acts as the data switch would act in a NAS airway facility. All of the information that the MicroVax II passes on can be recorded for later use.

Data Acquisition
One of the key design features of the system is the ability to acquire over 150 parameters from each of the flight simulators and record the flight performance of the pilots on an individual basis. This is important since an individual may require the ability to review the flight at a later date. Future enhancements to the system include the ability to replay a flight with the data collected

and actually have it drive the system, or some other desk top computer, which would enable the pilot to review not only performance and flight plan but flight path accompanied by an audio track in real-time.

Video and Audio
The ATCSS has three (3) video channels and twenty-four (24) audio channels available to it. Currently, three video and four audio channels are in use. The three video channels represent a recording medium for meteorological information, air traffic control, and a single aircraft participating in the system at a given time. The audio channels are broad band intercoms which operate as Channel No. 1 and Channel No. 2 for piloted vehicles, Channel No. 3 for inter-facility communications, and Channel No. 4 for supplemental information between the weather facility, simulation control, central processing, and the simulation manager. Future enhancements include actual frequency identification by piloted vehicles as well as video recording and biomedical recording for all personnel participating in the simulation.

APPLICATIONS
Flight Operations
In a traditional flight simulation, pseudo-pilots, or multiple piloted vehicles operated by one workstation, would be utilized to provide target information for the air traffic control environment. Similarly, the piloted vehicle might receive air traffic services from an instructor pilot or other pilot participating in the simulation. A typical scenario of ATCSS might happen as follows:

"A pilot would desire to participate in the simulation and would look on the schedule to find out when the next simulation is. The schedule would indicate that the next simulation would be the Washington, D.C. terminal airspace and the associated terminal control areas at 1400 on Friday. The simulation is scheduled for two hours and the pilot estimates that ten or twelve approaches can be practiced in that period of time. At 1330 the pilot would pick up the dedicated intercom telephone in the Flight Simulator Bay, contact the Flight Service Station, which is the meteorological laboratory within the confines of the ATCSS facility, and file a flight plan. The information required on this flight plan would be the same as that required by the FAA in order to perform in the operational environment. The student would be given a clearance void time and

a position for the simulator on the field. The student would then retrieve the maintenance board and keys for the simulator, go to the simulator room, turn on the simulator, pre-position the simulator as indicated by Flight Service, plug in the aviation headset, take out the National Oceanic and Atmospheric Administration (NOS) charts, and make a radio call to National Ground. At this time the controller in the ATC facility would call back and tell the student to contact the tower when ready for departure at Runway 18. The student would continue to pre-flight the simulator, receive his final checks, contact the tower, and call ready to depart Runway 18. The ATC facility tower controller would then respond and, on his tower visual system, would be able to see the simulator represented as an icon on a graphical projected image positioned at Runway 18. The controller would then clear the simulator for takeoff, the flight strip would become active, and the pilot would now be airborne".

ATC Operations

Using the same scenario, typically the flight strip would show up (when the student files his flight plan in electronic format) in the pending bay of the tower controller, or Local 1 controller, in the Washington National Airport tower simulator. Once the student has called ready for takeoff the controller would activate the strip. The strip would also show up in the pending bay at departure. Departure would await the handoff, activate the strip and work the traffic. All this time the strip is available to others who might have the target and would want the strip. In a traditional handoff mode the strips are physically passed or, in the case of long distances, electronically passed. In this scenario the strips are electronically displayed and are available as long as the aircraft target is displayed on the controller's screen. Controllers have the ability to operate not only two-dimensional coordinate sectors, but layered sectors based on altitude. This is a new method of operating electronically, hence the system automatically knows which aircraft are appropriate for that sector and is able to space primary and secondary targets as appropriate. Additionally, targets which are not appropriate are masked automatically unless the controller selects to view those targets. A third feature called shadowing allows the controller to rotate the scope to a three-dimensional display mode with the ground map and shadow to show latitude and longitudinal position and the actual target and data

block representing the altitude position. Additional ATC information is available in the form of velocity vectors and predictive information.

Flight Service Station

The ASSL at Embry-Riddle Aeronautical University (ERAU) has an established meteorological curriculum which includes a full service flight service station in a laboratory. The availability of this data has allowed the ATCSS to include not only updated weather in a real-time format, as available across the system, but also the ability to re-program meteorological phenomenon as required for the system.

An example of this is that the pilot operating in the Washington, D.C. area at 1400 on Friday might encounter a clear day with visibilities as great as six or seven miles. The likelihood that this would provide a realistic training environment without the student wearing a hood in a Frasca flight simulator would dictate that the simulation manager would probably ask for a more traditional day, something in the neighborhood of 700-800 scattered to broken, visibility 1-2 miles, or maybe even select a winter day with an overcast ceiling of 200 feet and 3 miles.

The Flight Service Station component also contains a PC program with a user-friendly format to file a flight plan. A pilot calling in over the intercom phone would be able to file a flight plan, and a student operating in this laboratory would simply enter the flight plan into the system, similar to the method in which it is done in the National Airspace System.

Software Design

The ATCSS project and the related Complex Team Simulation Training (CTST) project (funded by the Florida High Technology and Industry Council) have been formidable vehicles for interdisciplinary cooperation and development of aviation expertise by Aviation Computer Science students and faculty. Two software subsystems of the presented training platform have been designed and implemented in-house as an effort of cooperation between the faculty and students of the Aviation Computer Science Department and the Aeronautical Science Department. These subsystems are the Data Acquisition and Management Subsystem (DAMS) and the Air Traffic Control Subsystem (ATCS).

The DAMS is implemented on a MicroVax II minicomputer under the DEC VMS operating system using VAX FORTRAN as a

development programming tool. The subsystem consists of several modules representing the following functionality: a) acquisition of data from flight simulators, b) flight service station data entry, c) acquisition and preprocessing of the weather data, d) memory management, e) communication with the ATC subsystem, and f) subsystem control and utilities.

The ATCS is implemented on a network of Silicon Graphics (SG) workstations under UNIX with an IRIS graphics package, using C as a development programming tool. The modules of the ATCS represent the following functionality: a) background graphics with the area data base, b) foreground graphics with moving objects, c) communication with the data acquisition subsystem, d) multi-station broadcasting, and e) scope and program control.

The data from devices in physical proximity are acquired via serial cables and ports. The remote data are acquired using a multiplexer and fiber optic link. Experiments were conducted using a modem for the remote data acquisition. In the current version the communication between the MicroVax II and the Silicon Graphics (SG) network is accomplished by using a serial communication between the MicroVax II and one of the SG stations. The data from this station is then broadcast to the entire Silicon Graphics network. Recently, the TCP/IP protocol has been acquired and installed on the MicroVax II. Thus, the next phase of the project will allow the elimination of a serial link between the VMS and UNIX systems. Since the two systems are currently on an EtherNet, the broadcasting will be done from the MicroVax II directly.

The MicroVax II was chosen since it provides an excellent input/output capability. The number of available ports and large memory for storing data were the main decision factors. Silicon Graphics was chosen because of its excellent graphics capabilities, high speed, and networking protocols.

Management

The system is managed by a graduate student with experience in either flight operations or air traffic control. Traditionally, these students provide for continuity of the simulation system and assist in the overall functionality. Specific examples of actions that might be required, such as a controller who has an aircraft that is operating in the system and has requested something that the controller is not capable of handling, would be directed to the simulation manager. It is important to note that in this simulation environment both the pilots and the controllers are operating with less than full performance level certification or FAA certification. Therefore, part of the learning environment is the fact that mistakes will be made by the participants and in the teamed environment the participants will be required to resolve these problems. A good example is a pilot having some difficulty flying an ILS and who has requested a precision approach radar approach or vectoring to an airport that is VFR. At this time the simulation manager will designate a VFR airport and the controller would have to work this traffic through the problem. In a reverse scenario, a controller may have more traffic than can be managed effectively and the pilots would have to participate in the simulation on a basis in which the controller could manage the traffic, thereby receiving reduced service, extended flight periods, or other traditional problems which occur in the ATC system when it becomes taxed.

HUMAN FACTORS
Data Collection

All of the previously discussed applications highlight the fact that the human element was the primary design consideration in building the ATCSS. It was an underlying concept that the ATCSS platform would provide an excellent method of collecting empirical performance data on individuals teamed in an operational environment. Video availability, coupled with the audio tracks and future biomedical measurements, offer a platform that could become a viable tool for performance data collection. One of the unique features of this system is that all data is in digital format providing for a relatively simple retrieval and analysis process. It also provides a baseline for comparison between the piloted vehicles or between the air traffic control stations.

On-the-job Training

It was quickly realized when the ATCSS prototype became operational for the first time in the Fall of 1988 that the platform could provide an opportunity for on-the-job training for air traffic controllers. Originally it was conceived to provide training in instrument approaches principally for pilots. Both of these ideas have been pursued to the extent that it is now possible to mimic any given airspace both visually and mechanically. This provides an excellent opportunity for the system to

be used in educational environments.

TECHNICAL PROBLEMS
Software Implementation

There are two critical technical problems related to the practical application of computer science methodologies. The MicroVax II had to provide for orderly and timely acquisition of data and the subsequent transmission for use in the ATC subsystem. The Silicon Graphics network had to distribute the data over the network and provide a mechanism for communication and interaction between different workstations.

The first problem deals with the issue of real-time data acquisition and management. Several processes (independent programs) are run concurrently under VMS operating system. One such process is acquiring data from flight simulators (weather station, flight service). The data are then sent to the SG network. There is an evident need to provide a mechanism to protect data from corruption and provide for proper synchronization. This problem has been solved by using the concept of shared memory and the principle of mutual exclusion for the critical section of code in the modules manipulating the same data.

The second problem is related to the issue of inter-network communication. After trying various concepts, the concept of broadcasting packets of data through EtherNet using the socket mechanism was used to simplify communications. The transmitting station is broadcasting the available data and the receiving stations are picking only the data required. As mentioned above, the broadcasting, due to connection of the MicroVax II to the EtherNet, will be done from the MicroVax II.

Mode S

Development of the system was originally intended to be based on the Mode S standard of fifty-six (56) or 112 kilobits per second. This burst concept of data transmission was selected because it matches the Mode S transport requirement; however, in the form of serial communication this was abandoned simply because the technology of the facility did not provide an interface which would operate in these ranges. Otherwise, the design of the simulated ATC facility has remained relatively constant with the Mode S standard.

Distributed Participants

Another large technical problem looming over the ATCSS is the fact that persons have requested the opportunity to participate in the system from distances as far away as Atlantic City, New Jersey and Prescott, Arizona. Both of these distances present an additional problem in the form of transport delay incurred by the system in sending video and audio in real-time over these distances. Technology does exist that would allow this to take place; however, at this time, the control of the distributed participants would be limited by the central processor and by the system design itself. A window has been made for satellite communications or external fiber optic communications and currently the development team is testing an external fiber optic link to determine if that would be suitable for relatively short distances where fiber optic cable is available.

SUMMARY

Traditionally, flight personnel and air traffic controllers are trained independently of one another and interact only in the live operational environment. By engineering air traffic control stations and networking software to commercially available meteorological computers and flight simulators, a three-dimensional real-time airspace was created. The purpose of developing a multi-aircraft simulation was not to increase the level of fidelity of aviation training, but to provide a simulated environment that permitted forms of training otherwise not available in a simulated environment or feasible in an operational setting.

IMPROVING THE PERFORMANCE OF FLIGHT SIMULATORS VIA SMART I/O INTERFACE SYSTEMS

W. H. Deiss, E. J. Fadden, and R. M. Howe[*]
Applied Dynamics International
Ann Arbor, Michigan 48108

Abstract

This paper shows how the dynamic performance of flight simulators can be improved significantly through the use of multirate I/O as well as reordering of the computational code. The code reordering permits generation of simulation outputs before they occur in real time. This in turn permits interpolation/extrapolation to be used to generate multirate drive signals for DACs, including compensation for the half-frame delay normally associated with DAC outputs. Alternatively, the multirate outputs can be used to drive other digital devices or simulations operating with different frame rates or even asychronously. Multirate sampling and averaging of real-time inputs can also be used to improve simulation fidelity and eliminate aliasing. The Applied Dynamics SIMsystem is ideally suited to implement the concept of multirate I/O. Key hardware and software characteristics of the SIMsystem are described, and examples illustrating the implementation of these ideas on the SIMsystem are included.

Part I Multirate I/O

In real-time flight simulators the airframe equations of motion are integrated numerically using a fixed integration frame size h, i.e., an integration frame rate of $1/h$. Usually the real-time inputs are sampled at the same rate $1/h$ and the real-time outputs are also provided at the rate $1/h$. The outputs are normally converted to continuous signals with DACs (digital-to-analog convertors) using zero-order extrapolation.

The fidelity of the continuous outputs can be improved significantly by driving the DACs with a multirate output mechanized through digital extrapolation. This reduces the discontinuities between successive DAC outputs, since the DACs are being updated every h/N seconds, where the rate-multiple N can be made quite large. Digital extrapolation can also be used to compensate for the half-frame delay associated with zero-order DACs. The accuracy of the extrapolation can

be increased appreciably by providing inputs to the extrapolator algorithms before they occur in real time. In this case, the multirate outputs are in fact generated using combined interpolation/extrapolation.

In the next section, we will see how reordering of the computational code can be used to produce simulation outputs before they occur in real time. The following section then describes various interpolation/extrapolation algorithms and analyzes their dynamic accuracy, as well as overall accuracy when combined with DACs to produce continuous real-time outputs. Multirate real-time input sampling is then introduced. The multirate input average over each integration step h is used in the numerical integration of the aircraft equations of motion. This produces significant improvements in overall dynamic fidelity and also eliminates aliasing effects caused by input components with frequencies above one-half the base integration frame rate of $1/h$. Application of the above multirate I/O techniques to an example airframe simulation is then described. Also described is utilization of the gradient of aircraft acceleration components with respect to each real-time input, obtained as a by-product of multivariable function generation, for quasi-linear multirate integration of the aircraft equations of motion between main integration steps of duration h.

Part II of this paper describes the SIMsystem, a new family of hardware and software products from Applied Dynamics International designed for real-time simulation applications. A key component of the SIMsystem is the Applied Dynamics Real-Time Station (AD RTS). Motorola MC68040 microprocessors embedded within the I/O structure of the AD RTS permit implementation of the above multirate I/O formulas without increasing the computational load on the main flight simulation computer. Part III presents two examples illustrating how the ideas described in Part I are implemented in the SIMsystem.

---

[*]Associate Fellow, AIAA

Rearranging Computational Order to Produce Pure Time Leads

We begin this section with a review of the process of digital simulation using real-time integration methods. The state vector equations of motion for an airframe can be written in the following form:

$$\dot{V} = A[D, V, U(t)], \quad \dot{D} = V \ . \quad (1)$$

Here D, V, and A represent translational and rotational vectors of airframe displacement, velocity, and acceleration, respectively, and $U(t)$ is the real-time input vector, for example, control inputs by a pilot in a man-in-the-loop simulation. When the airframe is simulated in real time, a fixed integration step size h is used, since a variable step size is not compatible with real-time requirements. Thus the input and state vector components are represented at equally spaced discrete times. The output state vectors D_{n+1} and V_{n+1} at the $n+1$ frame, i.e., at $t = (n+1)h$, are computed from D_n and V_n at the nth frame using a real-time integration algorithm. The most popular real-time algorithm is the second-order Adams-Bashforth predictor method known as AB2. This is because AB2 is a second-order method (dynamic errors proportional to h^2, where h is the integration step size), requires only one evaluation of the state-variable derivatives per integration step, and is compatible with real-time inputs. When AB2 integration is used, the difference equations which are solved by the simulation computer are given by

$$V_{n+1} = V_n + h(\frac{3}{2}A_n - \frac{1}{2}A_{n-1}) , \quad (2)$$

$$D_{n+1} = D_n + h(\frac{3}{2}V_n - \frac{1}{2}V_{n-1}) , \quad (3)$$

where

$$A_n = A(D_n, V_n, U_n) \ . \quad (4)$$

At the beginning of the nth integration frame, i.e., at $t = nh$, the acceleration A_n is computed from the real-time input U_n, which has just become available, and the vectors D_n and V_n in accordance with Eq. (4). Then Eqs. (2) and (3) are used to calculate V_{n+1} and D_{n+1}. The simulation computer must be fast enough to complete all of these calculations in less than h seconds so that the states D_{n+1} and V_{n+1}, or any required function of these states, can be furnished as a real-time output at $t = (n+1)h$. It should be noted that in most airframe simulations the calculation of A_n in Eq. (4) requires much more computational time than the calculation of V_{n+1} and D_{n+1} in Eqs. (2) and (3). This

is because A_n usually involves very complex nonlinear functions, including multivariable aerodynamic functions that are evaluated by table lookup and linear interpolation.

We now examine the latency problem associated with AB2 integration. To do this we consider an input $U(t)$ which undergoes a sudden change and determine the elapsed time before this change is reflected in the simulation output displacement $D(t)$. We note first that the input $U(t)$ is sampled at integer frame times nh. This in turn means that the sample associated with a sudden change in $U(t)$ may be delayed for up to h seconds if the change occurs just after a previous sample. Eq. (2) shows that an additional one-frame delay occurs before the input sample U_n, which affects A_n, produces a change in the velocity V_{n+1}. Eq. (3) shows that this change occurs in the displacement D_{n+2}, a further one-frame delay. Thus the latency inherent in the AB2 method when applied to the airframe equations given by (1) is between $2h$ and $3h$ seconds.

The above latency can be reduced by one frame (h seconds) if trapezoidal integration rather than AB2 integration is used to obtain displacement from velocity. In this case Eq. (3) is replaced by

$$D_{n+1} = D_n + \frac{h}{2}(V_n + V_{n+1}) \ . \quad (5)$$

Here V_{n+1} must be computed from Eq. (2) prior to computing D_{n+1} from Eq. (5). Clearly a change in V_{n+1} causes a change in D_{n+1}, instead of requiring a wait for D_{n+2}, as in Eq. (3). It also turns out that using trapezoidal rather than AB2 to integrate velocity to displacement results in improved overall simulation accuracy, although the maximum allowable step size h compatible with numerical stability is reduced somewhat.[1]

One method for reducing the latency effects described above is to complete as many of the calculations associated with the nth integration frame as possible prior to nh, when the frame actually begins in real time. In mechanizing standard AB2 integration, as represented by Eqs. (2), (3), and (4), there is no reason why Eq. (3) cannot be implemented at the end of the $n-1$ integration frame, since it depends only on V_n and V_{n-1}, both of which are available at the end of the $n-1$ frame. This means that the next displacement state, D_{n+1}, is available at nh, one entire frame ahead of real time. The two-to-three frame latency associated with conventional AB2 integration is then reduced to between one and two frames. In an ongoing

simulation, this means that the output position state, within the accuracy of the AB2 algorithm, is actually available h seconds ahead of when it would occur in real time.

There may be other calculations required during the nth integration frame that can be completed prior to $t = nh$. In fact, the only calculations that cannot be started prior to time nh are those associated with the real-time input vector, U_n. Assume that these calculations take t_u seconds, and that the calculations which do not require U_n take $t_p = h - t_u$ seconds. Then the displacement D_{n+1} is available at time $t = (n+1)h - t_p$, i.e., t_p seconds ahead of real time. This procedure can be combined with early update of displacement D_n to D_{n+1} in AB2 integration, as described in the previous paragraph, to produce a total output displacement time lead of $h + t_p$ seconds, i.e., between h and $2h$ seconds.

Unfortunately, in typical airframe simulations, much of the complexity associated with the nonlinear acceleration function in Eq. (4) involves the input U_n. For example, the pitch-axis component of acceleration depends on the aerodynamic pitching moment coefficient C_M. The coefficient C_M may in turn be a nonlinear function of angle of attack α, Mach number M, flap position δ_f, and elevator displacement δ_e, where δ_e is a component of the input vector U that results from control-stick displacement. The four-variable function representing C_M is evaluated each integration frame by table lookup and linear interpolation, a process that is computationally intensive. In a typical airframe simulation, the time t_u required for calculation of multivariable functions involving real-time inputs can represent a substantial portion of the overall computation time for each integration step. As a result, the time $t_p = h - t_u$ available for advancing the output displacements in time, as described in the previous paragraph, may be somewhat limited.

However, there is a method to circumvent this limitation which leads to further potential improvements in the dynamic accuracy of simulations. In this method, the algorithm for evaluation of multivariable functions involving real-time inputs is executed so that linear interpolation with respect to the real-time input is performed last. In the above example for the pitching moment coefficient C_M, this means that interpolations with respect to α, M, and δ_f would be implemented first. This results in the calculation of the two quantities $C_M(\alpha, M, \delta_f, \delta_{e_k})$ and $C_M(\alpha, M, \delta_f, \delta_{e_{k+1}})$, where δ_{e_k} and $\delta_{e_{k+1}}$ are the data points within which the input δ_e falls. The final interpolation with respect to δ_e is accomplished using the formula

$$C_M(\alpha, M, \delta_f, \delta_e) = C_M(\alpha, M, \delta_f, \delta_{e_k}) + [C_{M_{\delta_e}}]_k (\delta_e - \delta_{e_k}),$$

where
$$(6)$$

$$[C_{M_{\delta_e}}]_k = \frac{C_M(\alpha, M, \delta_f, \delta_{e_{k+1}}) - C_M(\alpha, M, \delta_f, \delta_{e_k})}{\delta_{e_{k+1}} - \delta_{e_k}} .$$
$$(7)$$

Implementation of Eq. (6) requires only two additions and one multiplication, assuming $[C_{M_{\delta_e}}]_k$ has already been calculated. Of course, knowledge of the real-time input δ_e is necessary to determine the function table addresses for the data needed to accomplish the first three interpolations with respect to α, M, and δ_f, respectively, and hence to compute $[C_{M_{\delta_e}}]_k$. But suppose an estimate, $\hat{\delta}_e$, as might be obtained from extrapolation from previous δ_e values, is used to compute δ_{e_k}. Then this permits the computationally intensive calculation of $[C_{M_{\delta_e}}]_k$ to proceed without waiting for the real-time input δ_e. When the real-time δ_e finally arrives, only two additions and one multiplication, as noted above, are then required to complete the computation of the coefficient C_M in Eq. (6) and hence the pitch component of angular acceleration. This procedure is essentially equivalent to a local linearization of the acceleration function with respect to the input, where the linearization coefficient is updated each integration frame. As noted above, the procedure postpones the time when the real-time input is needed until t_u seconds before the end of the computational frame, where t_u can be small compared with the integration step size h. This in turn makes the frame output displacement available $h - t_u = t_p$ seconds ahead of its occurrence in real time.

Multirate Extrapolation and Interpolation

The output data sequence $\{r_n\}$ from a digital simulation can be converted from its basic data rate of $1/h$ to a multirate sequence with rate N/h, where N is a positive integer. This conversion process can be accomplished using either extrapolation or interpolation. In this section, we consider a number of different extrapolation/interpolation algorithms and note the dynamic accuracy associated with each method. Consider first extrapolation based on retaining the zeroth and first-order terms in a Taylor series representation of the extrapolation variable. If we define the extrapolation time interval as ah, where a is a dimensionless constant and h is the time-step size, then the continuous function estimate, $\hat{r}(t+ah)$, based on extrapolation using r_n and r_{n-1}, is given by the formula

$$\hat{r}(t + ah) = r_n + \frac{r_n - r_{n-1}}{h}(t + ah - nh) \ . \quad (8)$$

If we let r_{n+a} represent $\hat{r}[(n+a)h]$, then from Eq. (8) it follows that

$$r_{n+a} = (1 + a)r_n - ar_{n-1} \quad (9)$$

To obtain an equally spaced sequence of N data points over the interval $nh \le t \le (n+1)h$ in order to generate a fast data sequence with rate N times the rate $1/h$ of the original slow data sequence, we let a take on successively the values 0, $1/N$, $2/N$, ... , $(N-1)/N$.

The dynamic errors associated with a fast multirate data sequence generated from a slow data sequence using extrapolation can be analyzed in the frequency domain by letting the slow sequence be given by a sinusoid, $r_n = \sin \omega nh$. This analysis shows that the fast data sequence consists of a component at the input frequency ω, as well as components at $2(N-1)$ additional frequencies.[2] If ω_0 is the sample frequency of the slow data sequence ($\omega_0 = 2\pi/h$), these additional frequencies are equal to $\omega_0 \pm \omega$, $2\omega_0 \pm \omega$, $3\omega_0 \pm \omega$, ... , $(N-1)\omega_0 \pm \omega$. The fast data sequence output at the frequency ω will have very nearly the same amplitude and phase shift as the slow data sequence input of frequency ω. In fact, it can be shown that the amplitude and phase error of the output component at ω will be simply the average of the amplitude and phase errors, respectively, of the N extrapolator formulas used to generate the fast data sequence. For $\omega h \ll 1$, the fast data sequence components at the higher frequencies, $\omega_0 \pm \omega$, $2\omega_0 \pm \omega$, ... will have amplitudes that are small compared with the input amplitude.

In order to determine the amplitude and phase error associated with the fast data sequence, then, we need to determine first the amplitude and phase error for each of the N extrapolator formulas used to generate the sequence. For the first-order extrapolation represented by Eq. (9), it can be shown from z transform theory that the fractional gain error e_M and phase error e_A are given approximately by the following formulas:[2]

$$e_M(a) \cong \frac{a + a^2}{2}(\omega h)^2 \ , \quad \omega h \ll 1 \ , \quad (10)$$

$$e_A(a) \cong -\frac{a + 3a^2 + 2a^3}{6}(\omega h)^3 \ , \quad \omega h \ll 1 \ . \quad (11)$$

It follows that for extrapolation based on r_n and r_{n-1}, the predominant transfer function error is the gain error, which is proportional to the square of the step size h.

To determine the approximate gain and phase errors associated with the fundamental frequency component of the fast data sequence generated by extrapolation from the slow data sequence, we need to average the gain and phase errors for each of the N extrapolation intervals a. Using $e_M(a)$ and $e_A(a)$ as in Eqs. (10) and (11) to represent the gain and phase errors as a function of extrapolation interval, respectively, we then obtain the following formulas:

$$\begin{matrix} \text{Fast data sequence} \\ \text{gain error} \end{matrix} = e_{M_N} \cong \frac{1}{N}\sum_{k=0}^{N-1} e_M(k/N) \ , \quad (12)$$

$$\begin{matrix} \text{Fast data sequence} \\ \text{phase error} \end{matrix} = e_{A_N} \cong \frac{1}{N}\sum_{k=0}^{N-1} e_A(k/N) \ . \quad (13)$$

In the limit as $N \to \infty$ the summations in Eqs. (12) and (13) become the following integrals:

$$\begin{matrix} \text{Fast data sequence} \\ \text{gain error} \end{matrix} = e_{M_\infty} \cong \int_0^1 e_M(a)da \ , \quad (14)$$

$$\begin{matrix} \text{Fast data sequence} \\ \text{phase error} \end{matrix} = e_{A_\infty} \cong \int_0^1 e_A(a)da \ . \quad (15)$$

For a finite multirate ratio N the gain and phase errors are always less then those for $N=\infty$. Thus Eqs. (14) and (15) represent upper bounds for the gain and phase errors associated with a fast data sequence as obtained by extrapolation. Substituting Eqs. (10) and (11) into Eqs. (14) and (15), respectively, we find that $e_{M_\infty} \cong 5(\omega h)^2/12$ and $e_{A_\infty} \cong -(\omega h)^3/3$ for linear extrapolation.

For $N = 5$, Figure 1 shows the fast data sequence as derived from a slow data sequence, which consists of a portion of a sinusoidal cycle with $\omega h = 0.25$.

Often the simulation output r_n is a state variable, in which case $\dot{r}_n$ is also available and can be used in Eq. (8) instead of the approximation $(r_n - r_{n-1})/h$ to represent the time derivative. Then the extrapolation formula becomes

$$r_{n+a} = r_n + ah\dot{r}_n \ . \quad (16)$$

Here the extrapolator transfer function is given simply by $1 + j\omega ah$. It can be shown that the corresponding transfer function errors are equal to [2]

$$e_M(a) \cong \frac{a^2}{2} (\omega h)^2, \quad e_A(a) \cong -\frac{a^3}{3} (\omega h)^3, \quad \omega h \ll 1 . \quad (17)$$

Substituting these formulas into Eqs. (14) and (15), we find that $e_{M_\infty} \cong (\omega h)^2/6$ and $e_{A_\infty} \cong -(\omega h)^3/12$ for linear extrapolation based on r_n and $\dot{r}_n$. These errors are 0.4 and 0.25 times, respectively, the gain and phase errors found earlier for linear extrapolation based on r_n and r_{n-1}. We conclude that if $\dot{r}_n$ is available for extrapolation, it should be used.
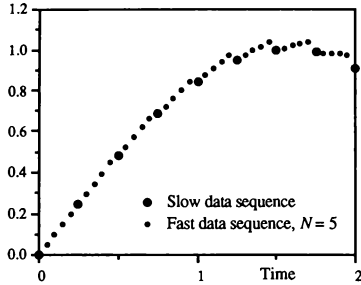
Figure 1. Linear Extrapolation ($N = 5$) Based on r_n, r_{n-1}

Figure 2 shows the fast data sequence as derived from a slow data sequence using extrapolation based on r_n and $\dot{r}_n$. The multirate ratio $N = 5$ and the slow data sequence is the same segment of a sinusoid used previously in Figure 1 for extrapolation based on r_n and r_{n-1}. Comparison of the figures shows the improved accuracy attained using r_n and $\dot{r}_n$ extrapolation.

For the two extrapolation algorithms considered thus far, the dimensionless extrapolation interval a has the range $0 < a < 1$. We have seen in "Rearranging Computational Order to Produce Pure Time Leads" how the order of computation in the numerical integration of the flight equations can be rearranged over each frame so that data points can be computed prior to their availability in real time. In fact, we have seen that r_{n+1} can be made available at the beginning of the nth frame by simply performing AB2 integration of the velocity $\dot{r}_n$ to obtain r_{n+1} from r_n at the end of the $n-1$ frame. If r_{n+1} is indeed available at the beginning of the nth frame, then we can use linear interpolation instead of extrapolation to obtain the multirate fast data sequence. In this case, the counterpart of Eq. (9) becomes

$$r_{n+a} = (1-a)r_n + ar_{n+1} . \quad (18)$$
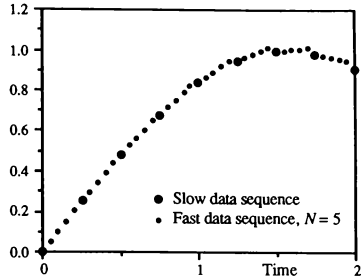
Figure 2. Linear Extrapolation ($N = 5$) Based on r_n, $\dot{r}_n$

As before, the dimensionless interval a has the range $0 < a < 1$. However, it is easy to see that this is equivalent to using Eq. (9) with a ranging from -1 to 0 instead of 0 to 1. Thus the formulas for extrapolator transfer function gain and phase errors can be converted to gain and phase errors for interpolation by replacing a with $a-1$.

If the data points from the simulation are available, say, only one-half frame ahead of real time, then the extrapolation interval a ranges from -1/2 to +1/2. Again, the transfer function gain and phase errors are obtained by replacing a with $a-1/2$ in the previous gain and phase error formulas for extrapolation. In any case, reducing the positive range of a lowers substantially the dynamic errors in generating the multirate data sequences, as we shall see later.

Further improvements in accuracy can be attained using higher order extrapolation algorithms. Table 1 summarizes the formulas for a number of extrapolator methods, including the two linear algorithms considered above, as well as two quadratic and one cubic algorithm. The table also shows the overall extrapolator/interpolator gain and phase errors for the multirate ratio $N = \infty$, which is equivalent to continuous extrapolation but very nearly correct for N larger than 3 or 4. For $0 < a < 1$, the errors are representative of pure extrapolation. For $-1/2 < a < 1/2$, the errors represent combined extrapolation/interpolation, where it is

97

Table 1
Summary of Multirate Extrapolator/Interpolator Gain and Phase Errors

Note: N = frame multiple = ∞ (equivalent to continuous output)

a = dimensionless extrapolation interval

All formulas are approximate based on $\omega h \ll 1$

Formula Inputs	Extrapolation Formula	0 < a < 1 Gain Error e_M	0 < a < 1 Phase Error e_A	-1/2 < a < 1/2 Gain Error e_M	-1/2 < a < 1/2 Phase Error e_A
r_n, r_{n-1}	$r_{n+a} = (1+a)r_n - ar_{n-1}$	$\frac{5}{12}(\omega h)^2$	$-\frac{1}{3}(\omega h)^3$	$\frac{1}{24}(\omega h)^2$	$-\frac{1}{24}(\omega h)^3$
$r_n, \dot r_n$	$r_{n+a} = r_n + ah\dot r_n$	$\frac{1}{6}(\omega h)^2$	$-\frac{1}{12}(\omega h)^3$	$\frac{1}{24}(\omega h)^2$	$0\cdot(\omega h)^3$
r_n, r_{n-1}, r_{n-2}	$r_{n+a} = \frac{2+3a+a^2}{2}r_n - (2a+a^2)r_{n-1} + \frac{a+a^2}{2}r_{n-2}$	$\frac{29}{60}(\omega h)^4$	$-\frac{3}{8}(\omega h)^3$	$\frac{103}{1920}(\omega h)^4$	$-\frac{1}{24}(\omega h)^3$
$r_n, r_{n-1}, \dot r_n$	$r_{n+a} = (1-a^2)r_n + a^2 r_{n-1} + (a+a^2)h\dot r_n$	$\frac{29}{360}(\omega h)^4$	$-\frac{7}{72}(\omega h)^3$	$\frac{29}{5760}(\omega h)^4$	$-\frac{1}{72}(\omega h)^3$
$r_n, r_{n-1}, \dot r_n, \dot r_{n-1}$	$r_{n+a} = (1-3a^2-2a^3)r_n + (3a^2+2a^3)r_{n-1} + ah(1+a)^2\dot r_n + a^2h(1+a)\dot r_{n-1}$	$-\frac{31}{720}(\omega h)^4$	$\frac{2}{45}(\omega h)^5$	$-\frac{23}{5760}(\omega h)^4$	$\frac{7}{2880}(\omega h)^5$

assumed the algorithm inputs are available one-half frame ($h/2$ seconds) ahead of real time in order to produce multirate outputs in real time. Note that the gain and phase errors are significantly smaller in this case compared with the errors for pure extrapolation ($0 < a < 1$).

Multirate Input Sampling and Averaging

In this section, we consider multirate sampling and averaging of real-time inputs. In order to better understand how this can be used to improve the accuracy of a real-time simulation, it is useful to introduce a modified form of the AB2 predictor algorithm. In this method, the acceleration vector A is evaluated at time $t = (n+1/2)h$ instead of nh, as in Eq. (4). To do this, it is necessary to have estimates of the velocity V, dis-placement D, and input U at the $n+1/2$ frame. These can be obtained by linear extrapolation using the n and $n-1$ frame values. Thus we let

$$\hat V_{n+1/2} = \frac{3}{2}V_n - \frac{1}{2}V_{n-1}, \quad \hat D_{n+1/2} = \frac{3}{2}D_n - \frac{1}{2}D_{n-1},$$
$$\hat U_{n+1/2} = \frac{3}{2}U_n - \frac{1}{2}D_{n-1}. \tag{19}$$

The difference equations for solving Eq. (1) are now given by

$$V_{n+1} = V_n + hA\left(\hat D_{n+1/2}, \hat V_{n+1/2}, \hat U_{n+1/2}\right), \tag{20}$$
$$D_{n+1} = D_n + h\hat V_{n+1/2}. \tag{21}$$

If the acceleration A is a linear function of D, V, and U, it is evident that this modified form of AB2 integration is exactly equivalent to the standard AB2 method represented by Eqs. (2), (3), and (4). However, when A is a nonlinear function of $D, V,$ and U, which is invariably the case in flight equations, the two methods give different results. In standard AB2 integration, first-order extrapolation is applied to the derivatives of the state variables, whereas in the above modified AB2 method, the first-order extrapolation is applied to the states. If the acceleration A contains discontinuous nonlinear functions, or if the input $U(t)$ has high frequency content, then the velocity state V will be a smoother function of time than the acceleration A. Under these conditions, which often occur, extrapolation based on the states instead of the state derivatives will give better results, and the modified AB2 method should be more accurate than the standard AB2 method. This indeed turns out to be true in typical nonlinear airframe and control system simulations.

Of course even better accuracy can be achieved by using $U_{n+1/2}$, the actual input at the $n+1/2$ frame,

instead of the estimate $\hat{U}_{n+1/2}$ given in Eq. (8). This does mean that an additional half-frame wait must occur before $U_{n+1/2}$ is available as a real-time input. But use of the technique of rearranging the order of computation and/or the local linearization of the acceleration function, as described in the previous section, should still permit the output state D_{n+1} to be computed in real time, or perhaps even slightly ahead of real time.

Yet a more accurate method for treating the input $U(t)$ in Eq. (20) is to utilize the average value of the input over the nth frame, $\bar{U}_{n,n+1}$, in place of $\hat{U}_{n+1/2}$ in Eq. (20). In fact, for the case where the acceleration A is simply equal to the input U (no dependence on D or V), use of the average input, $\bar{U}_{n,n+1}$, in Eq. (20) will give an exact result for V_{n+1}. This is because the average, $\bar{U}_{n,n+1}$, times the step size h is indeed the area under the U versus t curve. A numerical approximation to this average can be obtained by averaging N samples of the input U over the time interval from nh to $(n+1)h$. Thus we use the formula

$$\bar{U}_{n,n+1} = \frac{1}{N}(U_{.5/N} + U_{1.5/N} + \ldots + U_{(N-.5)/N})$$

or

$$\bar{U}_{n,n+1} = \frac{1}{N}\sum_{k=1}^{N} U_{(k-1/2)/N} \qquad (22)$$

Note that each of the N input samples is taken at the midpoint of the N intervals spanning the total sample period h. Of course, when this method of multirate input sampling is used, the computation of $\bar{U}_{n,n+1}$ cannot be completed until the last real-time input sample arrives, namely at $t=(n+1-1/2N)h$. This leaves only $1/2N$ seconds to complete the computation of the real-time output for the $n+1$ frame. It is still possible to do this by rearranging the computation order and using the scheme of local linearization of acceleration with respect to the input, as described in the previous section.

If N, the number of input samples averaged over each frame, is reasonably large, the latency variability of zero to h seconds associated with one input sample per frame is virtually eliminated.

Multirate I/O and Integration

In "Multirate Extrapolation and Interpolation" we saw how multirate outputs from a simulation can be generated using extrapolation, possibly combined with interpolation. These multirate signals can then

be used to drive DACs, which results in a much smoother continuous output from the simulation and an accompanying reduction in total dynamic error. In the previous section, we noted that multirate input sampling can also be used, where the average input over each integration frame h is used as the input to the main airframe simulation. Again, this improves overall dynamic accuracy, eliminates latencies associated with input sampling, and reduces aliasing effects.

Multirate input sampling can be combined with local linearization of the acceleration function with respect to input changes, as described in "Rearranging Computational Order to Produce Pure Time Leads," to provide an output data sequence at a frame rate which is a multiple N of the basic integration frame rate used in the airframe simulation. The multirate output data sequence can be obtained by simple numerical integration formulas if it is assumed that during each frame interval h the acceleration only changes as a result of changes in the input. N multirate integration steps are then performed over the nth frame using the multirate real-time input data. At the end of the N frames, the velocity and displacement states are updated to their new values, V_{n+1} and D_{n+1}, respectively, as computed in the main flight simulation with step size h. Further accuracy improvement in the multirate integration can be obtained by using linear extrapolation on the acceleration over the nth frame based on A_n and A_{n-1}.[3] When compared with multirate extrapolation/interpolation, the multirate integration scheme has the advantage of producing a multirate output which incorporates within each frame the effect of input changes during that frame.

A block diagram of the mechanization is shown in Figure 3. Here the computations associated with the multirate input averaging and the multirate integration are assumed to take place in the I/O interface system. In this way, there is no computational load added to the main flight simulation computer. The multirate output data sequence can be used to drive DACs when the required real-time outputs are continuous. Alternatively, the multirate output data sequence can be used to provide the input to a second digital simulation. If the multirate ratio N is made large, this second simulation will always receive a input with negligible quantization error in time. Thus the second digital system can be operating at a completely different and noncommensurate frame rate. It can even be an asynchronous digital system.
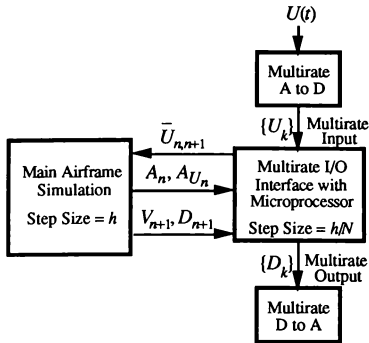
Figure 3. Multirate I/O, Including Multirate Integration, Using a Smart Interface

Example Simulation

In this section, we consider an example simulation, specifically, a business jet flying at 40,000 feet with a speed of Mach 0.7. We simulate the airframe response to the acceleration-limited pulse of elevator displacement shown in Figure 4. In order to demonstrate clearly the errors due to the step size used for numerical integration, we choose an unrealistically large step size of $h = 0.1$ second for the simulation.
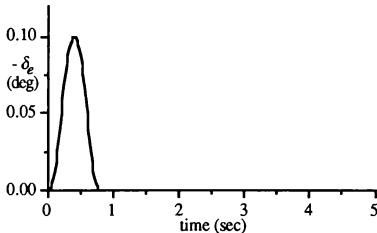
Figure 4. Acceleration-limited Pulse Input of Elevator Displacement

Figure 5 shows the simulated pitch angle response θ as obtained at the DAC output when AB2 integration is used. Also shown, is the ideal continuous solution. The dynamic errors due to the very large step size, as well as the startup latency inherent in AB2 integration, are clearly evident. Also evident is the one-half frame delay in the average DAC output.
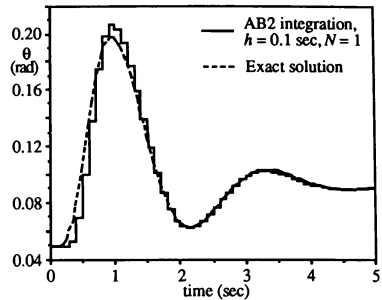
Figure 5. Airframe Pitch Angle at DAC Output When Using Conventional AB2 Integration, No Multirate I/O

Next, in Figure 6 we see the DAC output when multirate input sampling and averaging, along with multirate quasi-linear integration, as introduced in the previous section, is utilized. The multirate ratio $N = 4$ in the example shown still allows the step discontinuities of the DAC output to be discerned. Comparison with the result in Figure 5, where no multirate I/O and integration was used (i.e., $N = 1$) shows the substantial improvement in output accuracy and smoothness. Note that this improvement has resulted only from multirate I/O and integration, all of which can be mechanized at the interface with no added computational load on the main airframe simulation.
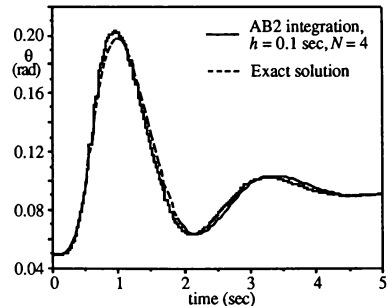
Figure 6. Airframe Pitch Angle at DAC Output with Conventional AB2 Integration and Multirate I/O and Integration, $N = 4$

Finally, in Figure 7 the DAC output is shown for $N = 4$ when modified Euler integration is used instead of AB2 integration.[3] If a multirate ratio N much larger than 4 had been used, the discontinuities in DAC output would barely have been discernable. The modified Euler method, where displacements and accelerations are defined at integer times and velocities at half-integer times, is particularly well suited to multirate input sampling and averaging.
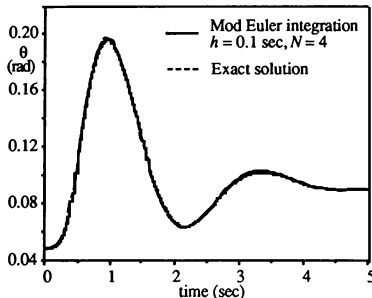
Figure 7. Airframe Pitch Angle at DAC Output Using Modified Euler Integration and Multirate I/O and Integration, $N = 4$

Part II The Applied Dynamics SIMsystem

Since 1957, Applied Dynamics International (ADI) has developed and produced specialized computer systems for real-time, hardware-in-the-loop (HITL)[*] simulation that were targeted at the high-performance end of the engineering test and evaluation market. To date, ADI products have only rarely been used in training simulator applications (although the first two FAA-approved helicopter trainers based on blade-element modeling of the main rotor used ADI computers). In general, ADI's products have been overkill performance-wise and too expensive for the training simulator market.

ADI has recently introduced a family of new hardware and software products denoted collectively as the SIMsystem. The SIMsystem represents a change in product thrust for ADI away from the special-architecture simulation "compute engine." ADI's emphasis now is on providing an integrated set of microprocessor-based hardware and software tools, including third-party products, to simplify the user's job of developing a high fidelity, validated, HITL simulation.

The SIMsystem is characterized by:

An open hardware and software architecture;

An expandable level of distributed computational resources;

Integral I/O[**] capabilities featuring automatic calibration, self-test and on-line test modes as appropriate;

Sophisticated communications and control mechanisms;

An integrated, network-based software infrastructure with a windowing, graphical user interface in which application packages and support for programming languages can be embedded;

Software for scheduling, synchronizing, and controlling the activities of distributed parallel processors;

A variety of software tools and program development packages such as FORTRAN, C, and Ada;

A low-level debugger and a comprehensive set of diagnostics; and

A workstation graphics package for generating strip-chart and X-Y plots of simulation variables in real time.

---

[*] As used in this paper, hardware-in-the-loop or HITL simulation involves modeling a physical system or subsystem on a computer and interfacing this computer model with hardware of some type. This hardware may be part of the system being studied and/or may be a mock-up of real system hardware to facilitate person-in-the-loop studies and training simulator applications.

[**] As used in this paper, I/O refers to the collection of interface devices required to provide the coupling between the computer and the user's external hardware, whatever that might be. It does not refer to the typical digital computer peripheral devices like disks, terminals, and printers.

The tightly coupled trends of ever-increasing performance and continually decreasing prices ensure that microprocessor-based compute power will soon be a commodity, if it isn't already. This necessitates a new way of thinking about how computers should be used in HITL simulation and training simulator systems. Before delving into an overview of the SIMsystem, it is useful to consider one philosophical approach to the architecture of a distributed system for high-fidelity HITL simulation that treats computer power as a commodity.

The concept is based on a decomposition of the overall HITL simulation task into a set of relatively independent subtasks by the user. Each of the subtasks is then assigned to an appropriate computational resource. Not surprisingly, the SIMsystem accommodates this approach, among others, very well.

First, a high-performance processor with adequate on-board memory to avoid any run-time disk accesses should be devoted to the *SOLE* task of solving the simulation model equations. Preferably, this computational unit would have no operating system. This may not be practical, but overhead due to any real-time kernel should be kept to an absolute minimum. Efforts should be made to maximize the integration frame rate.

Next, an intelligent I/O system should be used. All I/O-related computational tasks such as format conversion, scaling, digital filtering, boolean operations, and handling of interrupts should be assigned to processors in the I/O system. Predictive techniques for latency reduction should be implemented in these processors. Every effort should be made to eliminate or minimize software overheads.

Finally, additional computer resources should be used to handle activities of lesser time-criticality such as data logging, user 'bus snooping", etc., so as not to burden the processors involved in time-critical operations.

This approach seeks to maximize simulation system performance by throwing dedicated computer resources at tasks to be performed, as if cost were of no consequence. In the past this idea would have been ridiculous, but in the future this approach will be common place. ADI is taking advantage of these trends in microprocessor price and performance in the SIMsystem to provide an expandable level of parallel computer power.

The use of parallel computer resources in a training simulator can certainly improve performance for time-critical operations, particularly if the methods described in the previous sections in this paper are implemented. But as time goes on, the real reason for the liberal use of distributed compute power will prove to be an economic one; this is the route to lowering both engineering development and overall production costs.

Today's concern for the user is not so much how to obtain a very high level of compute power for a given application as it is to bring this compute power to bear on the application effectively and productively. Numerous vendors offer VMEbus-based single board computers that are easily integrated into the system hardware-wise. However, the system integration task involving the partioning of a large-scale simulation across a number of these computers is non-trivial. The lack of good system integration tools has clearly been a major problem. The SIMsystem addresses and handles this issue of systematizing the use of parallel intelligent resources.

The SIMsystem integrates loosely coupled, distributed computer resources with a broad range of I/O capabilities. These I/O capabilities include analog and discrete line input and output devices, interfaces to a variety of standard digital buses and computer I/O ports, and a family of hardware simulations of common transducers used to couple to a digital controller, e.g., a jet engine controller.

These powerful, parallel facilities are orchestrated to function as a finely tuned entity through the use of a sophisticated hardware/software communication and synchronization control system. The user's productivity in developing an HITL simulation as well as the run-time performance of the system were both carefully addressed in the design of the SIMsystem.

ADI's initial product entry with the SIMsystem is aimed primarily at its long-time, high-performance market niche. However, the structure of this new product line is far more versatile than that of ADI's earlier products, and the SIMsystem is easily extended to serve a much broader market base. In particular, ADI is adding products to this family that are designed specifically with training simulator price/performance and related (e.g., simulator cabling) considerations in mind.

Space does not permit a detailed discussion of all aspects of the SIMsystem in this paper. For more detail on the SIMsystem, see reference 4. The focus here is

on those facets that most directly impact the run-time performance and the user's development process. After all, as the cost of simulation system products decreases, the spotlight falls more and more on the user's engineering costs in developing, debugging, and validating an HITL simulation.

Features of the SIMsystem of particular interest for training simulator applications are:

The Applied Dynamics Real-Time Station (AD RTS), a VMEbus-based, local area network (LAN) resource accessible from user workstations operating under UNIX or VMS which includes distributed computer facilities, an integral I/O capability, and sophisticated communication/control mechanisms; and

COSIM, a software package which facilitates the use of parallel computational assets by allowing the user to define the structure of distributed processes, describe their communication and synchronization operations, and combine the pieces into a coherent unit.

The Applied Dynamics Real-Time Station

A block diagram of the AD RTS is shown in Figure 8. As noted above, the AD RTS is a server that attaches to an Ethernet-based LAN. User access is via workstations connected to the AD RTS through the LAN. The AD RTS can be used in stand-alone mode or in conjunction with another computer system such as one from DEC, Concurrent, Encore/Gould, or Harris.

VMEbus Interact Manager. The backbone of the AD RTS is its system VMEbus. The bridge between this bus and the LAN to which the AD RTS is attached is provided by the VMEbus Interact Manager or VIM as shown in Figure 8. The VIM, a Motorola MC68030-based single-board computer with a 100-Mbyte disk drive and an Ethernet port, plugs into the system VMEbus of the AD RTS and provides the interface for all user interaction with the various facilities of the AD RTS. The VIM:

Runs UNIX System V and manages all aspects of the network communication with the user's workstation;[*]

Loads and starts other AD RTS system components and supports operations such as debugging and running diagnostics; and

Handles all user interactive requests to the AD RTS at run-time and other run-time-related activities such as servicing data transfers for logging and graphing purposes that must be performed in a timely fashion, but are not highly time critical.

VMEbus Performance Considerations. Delays or latency in the movement of time-critical data can cause serious dynamic errors in an HITL simulation. In a bus-based system, such as the AD RTS, bus performance is a very important consideration. VMEbus characteristics are controlled by a detailed specification. The maximum potential performance of the VMEbus for block transfers of data over the 32-bit data portion of the bus is 40 Mbytes per second. VMEbus operation is asynchronous and the bus data rate actually achieved is essentially determined by devices on the bus and bus arbitration overhead. In a typical VMEbus system, bus performance can be expected to be well below the maximum possible rate.

Another issue is how effectively the bus is used in an application. The data rate achieved on the bus may be relatively high, but if much of the bus traffic is "churning" due to poorly organized system facilities and/or poor facility use, the overall performance will suffer.

Optimizing VMEbus performance in the AD RTS was a major design goal. The bus traffic that normally occurs in a general VME system is greatly reduced in the AD RTS and not allowed to interfere with the efficient movement of time-critical data. This is accomplished by using:

A hierarchical bus structure containing a primary or system VMEbus and optional VME Subsystem Buses (VSBs) and/or optional secondary VMEbuses; and

A VMEbus communication management scheme that tightly controls all time-critical data flow activities.

The Communication Processor. The vehicle for tightly controlling run-time communication traffic in the AD RTS is the Communication Processor, or COP. The COP provides a computer-based, low-latency

---

[*] Since the VIM functions in a UNIX setting, it does not play any role in activities that are time-critical in the sense that these activities impact the proper real-time behavior of the HITL simulation.

to external simulation
computer   to user's hardware

Fiber-optic
or
Ethernet
port

COP
(ADI)

Parallel
Intelligent
Resource
(ADI)   . . .   Parallel
Intelligent
Resource
(ADI)

VMEbus
Interact
Manager
(ADI)   to local
area network

System VMEbus

VMED   . . .   VMED   Bus-to-Bus Bridge

VMEbus

VMED   . . .   VMED

VMED = 3rd Party Device

Figure 8. Block Diagram of the AD RTS

interconnect and networking capability for all transmissions of time-critical data both within the AD RTS and between the AD RTS and an external computer. It controls all time-critical activities on the system VMEbus at run time in a way that maximizes bus efficiency. The COP uses an Advanced Micro Devices Am29000 microprocessor as its controller. The Am29000 is well-suited for this application where high-speed floating-point capability isn't needed, but a high level of performance is required for handling I/O operations, interrupts, and data manipulation.

The COP fills a variety of rolls in the AD RTS, including that of system controller.* In addition, the COP has a port to connect the AD RTS to an external simulation compute engine and/or another AD RTS. This port can be connected to either a fiber-optic or Ethernet link.

As will be discussed later, the COP performs gather/scatter operations as part of its run-time

communication control activities. The VMEbus data rate achieved in the AD RTS for these COP gather/scatter operations is approximately 16.7 Mbytes per second.

The VIM and the COP provide computer resources to handle special functional needs within the AD RTS. Neither of these devices takes a direct part in performing simulation- or I/O-related computational tasks at run time.

Parallel Intelligent Resource. In interfacing a simulation to hardware, the user is often faced with performing operations in the interface area that are best handled by a computer. These include interface control, data formatting (such as fixed to floating point), data scaling, bit packing, logical operations, digital filtering, handling local analog loops with high bandwidth, and controlling I/O-related subsystems. To the extent that these operations are performed locally in the interface without requiring VMEbus transactions, I/O traffic on the VMEbus is reduced, which is highly desirable.

---

* The VMEbus specification calls for a system controller, which performs special duties such as bus arbitration, to be located in slot one of the VMEbus backplane.

I/O operations can generally be divided into independent blocks (including associated computational tasks) that can be handled in parallel. This observation led to the concept of the Parallel Intelligent Resource, or PIR. A block diagram of a PIR is shown in Figure 9. Each PIR has a powerful, user-programmable, microprocessor-based Simulation Processor, or SP, which services one or more I/O function units. Space does not permit a description of the packaging methods used in the AD RTS. Suffice it to say that the SP is contained on a circuit board unit into which one I/O function unit can be plugged directly and up to five additional units can be connected through a VSB.

The I/O capability of an AD RTS can be expanded by the addition of a PIR or by adding to a partially expanded existing PIR. Price/performance trade-offs will generally influence the direction taken.

Every VMEbus device: a) must be capable of being a slave device on the bus for data read/write purposes; and b) may be, but does not have to be, capable of being a bus-master. The PIR is designed to operate only as a slave device on the VMEbus. All movements of data to/from the PIR are controlled externally; the PIR cannot generate any traffic on the VMEbus. The dual-ported memory allows the PIR and the system VMEbus to operate in parallel with neither one interfering with the other. Therefore, the PIR can execute very elaborate programs internally, without having any impact on the system VMEbus. This is in keeping with the stated design objective of maximizing system VMEbus performance.

The Simulation Processor. The microprocessor in each SP is the Motorola MC68040. SP options include an Ethernet port, a VSB interface, and a SRAM expansion memory of either 256 Kbytes or 1 Mbyte, as shown in Figure 9. The SP's MC68040 is loosely coupled to the system VMEbus through a 32-Kbyte, dual-ported memory. One port of this memory
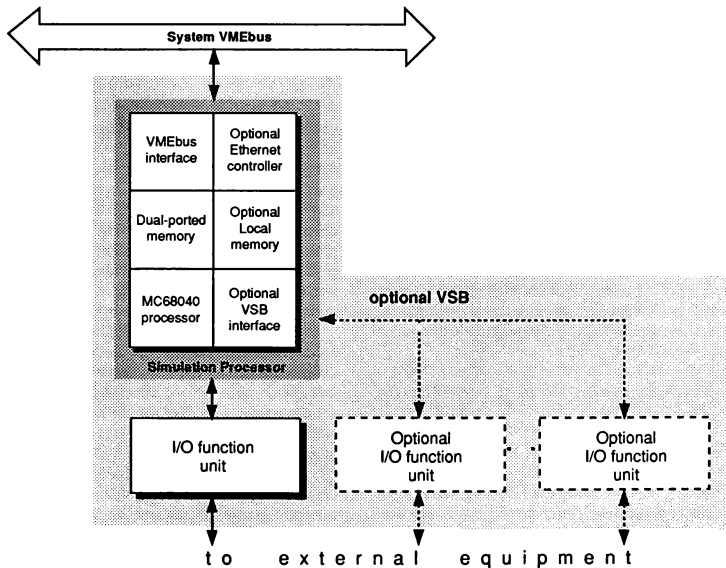
Figure 9. Block Diagram of a PIR

105

connects to the VMEbus interface; the other port connects to the MC68040. The MC68040 cannot initiate any data transfers to/from the system VMEbus and, hence, cannot serve as a bus master (but the MC68040 can access the VMEbus interrupt structure to send an interrupt to the system controller).

The PIR's combination of I/O devices with a powerful microprocessor makes it very simple for the user to implement a broad range of I/O-related computational activities, such as the techniques described in the first part of this paper. The microprocessor also readily accommodates convenient user features such as: automatic calibration of analog devices (accomplished without the need for trim potentiometers); and self-test and on-line test operations.

I/O Function Units. Field Programmable Gate Arrays (FPGAs) are used where appropriate in I/O function units. Each FPGA incorporates a block of Random Access Memory (RAM) for storing a program which configures the array of logic elements (nominally 4,000 gates) in the FPGA. The configuration of these elements is readily changed by writing a new program into the FPGA's RAM. The FPGAs augment the MC68040 in the associated SP by providing the "glue" logic needed to support specific I/O functionality.

Various I/O function units are available for the AD RTS and additional units will be developed on an ongoing basis. The units currently available provide: discrete or digital input/output line (i.e., sense line and control line) capability, 12- and 16-bit analog input/output capability, various types of standard digital device interfaces (e.g., DRV11-WA, IEEE-488, RS422, and RS232/423), and a collection of hardware simulations of common transducers such as resistance temperature devices (RTDs), linear variable differential transformers (LVDTs), etc.

The Digital Interface Module (DIM) provides the user with:

Four independent groups of 16 discrete I/O lines used for sensing or controlling external events. Each group of 16 lines can be configured as input lines only or output lines only or can be operated as a 16-bit bidirectional bus. If all four groups are configured for bidirectional operation, the user has a very flexible 64-bit bidirectional bus suitable for interfacing to external devices. A variety of configuration-dependent options are available to the user;

Eight special logic lines that can be used for strobes, handshake operations, external interrupts, etc.; and

One line consigned to each group of 16 I/O lines to accommodate a user-supplied high-rail voltage (up to +48 volts) for some of the available options.

Two different Analog I/O Modules (AIMs) are available, one with 12-bit devices and the other with 16-bit devices. The AIM provides the user with:

Eight analog-to-digital converters (ADCs) and eight digital-to-analog converters (DACs). The DACs and ADCs are totally independent devices, but are arranged in eight ADC/DAC pairs with a comprehensive switching scheme to facilitate automatic calibration, a self-testing mode, and an on-line test mode; and

Eight special logic lines that serve the same purposes as the special logic lines on the DIM, and one group of eight logic lines that can be used as input lines, output lines, or as a bidirectional bus.

Each DAC is double-buffered; DACs can be updated singly or in groups. Independent sensing of both the ground differential of the load relative to the DAC and the output voltage applied to the load (measured using an unloading amplifier) is provided to eliminate the effects of IR drop in the wiring and any offset voltage between the grounds. This remote sensing, together with the use of the on-line test feature, ensures that the desired output voltage is really applied across the load.

Each ADC has a differential input with remote ground sensing to eliminate any offset voltage between the grounds and an unloading amplifier in the primary input line to eliminate the effect of IR drop between the voltage source being read and the ADC. ADCs can be put in the convert mode either singly or in groups.

The AIM and the DIM provide the kinds of I/O capability typically associated with an HITL simulation system. In some applications, though, a more specialized form of I/O capability is required.

Consider, for example, the task of interfacing a "black box" digital engine controller to the simulation of a jet engine. The only access to the controller is through the connectors in the package that normally interface it, via cables, to the appropriate transducers

(sensors and actuators) located in various parts of the engine and to other controls and displays in the vehicle to which the engine is attached. Transducers typically involved in this kind of application include: thermocouples, resistance thermometer devices (RTDs), strain-gage pressure sensors, linear and rotary variable differential transformers (LVDTs and RVDTs), pulse-generating rotational speed sensors, synchros and resolvers for determining rotational angles, and torque motors. These transducers must be replaced by appropriate interface hardware in order to couple the real controller to the simulated engine.

For a number of sensors, the controller feeds the sensor with an excitation signal and receives back from the sensor a signal related to both the excitation signal and the variable being monitored. From the above list, this method of operation is used for RTDs, strain gages, LVDTs, RVDTs, synchros, and resolvers. For some devices, the excitation signal is DC; for others an AC signal is used. AC excitation frequencies often range from 100 Hz or less for some devices to 20 kHz and above for others. The interface hardware that simulates these sensors must be realistic to the point of providing both the appropriate input signals to the controller and the proper electrical load impedances to the controller's excitation signals.

Similarly, in the case of actuators, it is necessary to measure the controller's drive signal for use in the simulation and to load the drive source with the proper electrical impedance.

Signal amplitudes vary significantly depending on the type of device. The maximum output for a thermocouple may be 75 millivolts or less. Some transducers have outputs in the range of several hundred millivolts while others are in the range of several volts. Pulse-generating rotational speed transducers may have output signals that vary in both amplitude and frequency as a function of rotational speed with pulse frequencies in the 10 Hz to 50 kHz range or higher.

This specialized interface requirement is handled in the AD RTS through the use of the Transducer Interface Module (TIM). The TIM consists of a base module with provision for up to eight plug-on, "mezzanine" units. Each plug-on unit contains one or more transducer simulators. Since most of the devices to be simulated are analog, the transducer simulators generally use analog technology including such elements as operational amplifiers, 12-bit multiplying

DACs, etc. These transducer simulators incorporate both automatic calibration and test modes.

Digital device interfaces (e.g., DRV11-WA, IEEE-488, and RS-422) are also handled in a PIR in the AD RTS. The I/O function unit for such devices is the Bus Interface Module, or BIM. The BIM is similar to the TIM in that it consists of a base module with provision for plug-on mezzanine units. The various digital device interfaces are provided on mezzanine units. This approach is very flexible in terms of system configuration.

The digital device interfaces employ first-in-first-out (FIFO) memories and microcontrollers where appropriate to reduce the load on the PIR's SP while maintaining a high level of performance.

In addition to the facilities available through the various function units for the PIRs, the user can also incorporate third-party VMEbus devices (VMEDs) in the AD RTS, as indicated in Figure 8. The AD RTS packaging scheme allows VMEDs to be either 6U by 160 mm or 9U by 400 mm in size.

Training System-specific I/O Function Units

For the most part, the above I/O function units are targeted at the requirements of the engineering test and evaluation market. Training simulators have somewhat different and unique needs. These needs include a higher number of I/O points, lower sampling rates, and more widely distributed hardware interface points than are encountered in the typical engineering test and evaluation situation. ADI is presently developing I/O function units designed to meet these unique simulator requirements. In particular, as part of the design of these new I/O components, ADI is incorporating some innovative ideas for reducing cable runs.

COSIM

The AD RTS provides Parallel Intelligent Resources (PIRs) embedded in a sophisticated communications and control system directed by the Communications Processor, or COP. The COP does not take part in any computational task, but manages all time-critical activities at run time. The COP operates in parallel with all other processors to handle the movement of data between or among the other processors and to synchronize their activities as necessary. COSIM, the real-time executive for the AD RTS, is the vehicle by which the user:

Schedules and synchronizes the COP and other processors in the AD RTS so that they operate harmoniously in a time-efficient fashion; and

Interacts with the hardware capabilities of the AD RTS for programming and debugging purposes and for controlling the execution of an HITL simulation.

Run-time COP and SP programs are generated by COSIM. COSIM supports the parallel processor structure of the AD RTS by providing the "communications glue" for the various components of a distributed HITL simulation. It allows the user to define the structure of the distributed processes, describe their communication and synchronization operations, and combine the pieces into a coherent unit by giving the user the means for scheduling operations, synchronizing processes, and controlling interprocessor communications. COSIM supports both synchronous polling loops with explicit scheduling of processor operations and asychronous servicing of events using interrupt handlers. Explicit scheduling of operations eliminates task switching overheads which leads to very efficient run-time control of the parallel resources in the AD RTS. As will be seen, COSIM has both programming and operating system characteristics.

COSIM includes:

A flexible high-level language for programming the communications sections of distributed HITL simulations;

A cross-compiler that runs on the user's workstation and generates code for the COP and other processors in the AD RTS;

An extensive run-time library that supports a wide range of I/O-related operations;

A set of program templates to minimize program development effort; and

A highly interactive framework, COSIM Interact, for handling off-line activities such as system setup and debugging and for controlling and monitoring all run-time aspects of the HITL simulation.

Run-Time Communication Control. In the most elementary instance of an HITL simulation, one cyclic repetition rate or frame rate suffices for the

numerical integration of the model equations and all other operations that have to be performed. For the sake of simplicity, a single-frame-rate situation with no external interrupts serves as the basis for the following discussion.

The frame rate for the AD RTS can be established in the following ways: by a system linked to the external port of the COP, through the use of a programmable timer in the COP, by an external computer or other device connected to an I/O interface, or by another device, such as a VMED, contained within the AD RTS. The specific method for establishing the frame rate is incorporated in the COSIM program.

Once the basic frame rate is established, it is necessary to schedule and interrelate the numerous activities that must occur within a frame. These activities can be grouped in three mutually exclusive categories which relate to the COP's operations. In the following discussion, these categories are denoted as *primetime, freetime*, and *showtime* activities.

These three types of activities may be interspersed very flexibly within a frame. However, the easiest approach to understanding the communication control concept used in the AD RTS is to think of a frame as divided into three subperiods of unequal length where:

Primetime is the "critical time" period during which all time-critical data movements take place;

Freetime is a noncritical interval that is "free" for other activities (e.g., communications between or among VMEDs) to take place, except for activities that involve the VIM; and

Showtime is an interval devoted to servicing any user interactive "show me" and related types of requests posted through the VIM including such operations as "bus snooping," data logging, parameter changing, performance profiling, etc.

During primetime, the AD RTS can be viewed as operating in a star configuration with the COP at the hub. The COP takes control of the VMEbus and serves as bus master, locking out all other devices. This is accomplished by giving the COP a higher bus-request priority than any other device on the VMEbus. In this interval, the COP performs gather/scatter scan operations:

To move data received by the COP through its external port to PIRs, VMEDs, and secondary VMEbus subsystems in the AD RTS;

To move data from PIRs, VMEDs, and secondary VMEbus subsystems in the AD RTS to the COP for transmission from the external port on the COP; and

To move data from one PIR, VMED, or secondary VMEbus subsystem to another within the AD RTS.

The gather/scatter scan operations are organized by the COSIM software when it generates the COP program. These gather/scatter scan operations feature pipelined addressing to make very efficient use of the VMEbus bandwidth. The communications handled by the COP may be synchronous, semisynchronous, or asynchronous.

Issues relating to the synchronization of time-critical operations are handled as part of the primetime activity. Synchronization of AD RTS operations with simulation computations taking place in a computer tied to the AD RTS via the external port on the COP, can also be handled by COSIM.

During freetime, the COP relinquishes its control of the VMEbus. VMEDs which plug directly into the system VMEbus and are capable of operating as a VMEbus master then have access to this bus. It is expected that these devices will be used for operations that are not particularly demanding in terms of speed and where it is not critical that operations requiring communications via the VMEbus be completed in each integration frame.* Note: A processor-based VMED may be needed to manage the operations of any nonprocessor VMEDs. Depending on the specific user requirements placed on the VMEDs, it may be desirable to have this VMED processor run some suitable real-time kernel.

During showtime, the COP again takes command of the VMEbus and responds to the user's interactive requests for services such as bus snooping, data logging, etc., which have been placed with the VIM. In normal operation, the VIM is a slave device on the

* Proper system operation under COSIM is based on the premise that any VMED will have a lower VMEbus request priority than the COP and that a VMED will not unduly maintain control of the VMEbus to the point of interfering with the normal, COSIM-directed COP functions. However, it is the user's responsibility to ensure this; COSIM does not impose any constraints on the user in this regard.

VMEbus during an HITL simulation run. (However, the VIM is capable of taking control of the AD RTS during run-time under certain circumstances, such as the issuance of a user "hold" command.) This is the vehicle used for isolating the UNIX environment of the VIM from the real-time activities of the rest of the AD RTS. The VIM signals the COP that there is a user request for service by the "mailbox" technique of placing a message in a designated part of its memory space. During showtime, the COP checks its VIM mailbox for messages and carries out those activities that it can in the remaining part of the showtime period. Activities not completed in one showtime period are carried over to the next one, since these are not time-critical in nature.

To summarize, during an HITL simulation run, the COP via its COSIM program provides:

Run-time control of its external port packet controller through commands and interrupts;

Control of VMEbus mastership, thus managing all critical timing operations and optimizing data throughput;

Control of gather/scatter transfers of time-critical data through use of the gather/scatter scan table;

Data transfer management for all PIRs and some or all of the VMEDs; and

Service of requests (e.g., snooping, logging, and performance profiling operations) from the VIM.

COSIM takes advantage of the highly efficient communications architecture of the AD RTS by operating at the system level to provide the structure required to coordinate and interrelate activities on multiple processors. A master/slave mode of operation is implemented in COSIM with the COP acting as master during primetime and showtime activities.

The COSIM Language. COSIM is a very versatile high-level language for:

Specifying the distribution of programming tasks among the various parallel computational resources;

Defining the scheduling structure, including synchronization requirements for parallel computational operations;

Programming the communications sections of each of these parallel operations; and

Supporting all of the various I/O and digital interface capabilities of the AD RTS through the use of an extensive run-time library.

(Note: COSIM is not used for programming the computational operations that take place in the parallel processors. The computations are programmed in one of the available programming languages such as FORTRAN, C, Ada, or Assembler and linked with the COSIM program.)

While COSIM was designed to support the facilities of the AD RTS, it is essentially machine independent. Hardware-specific activities are handled via library routines.

The COSIM language is small in size and employs regular, modern syntax that is akin to the Pascal programming language in many respects. Readability was emphasized in its design. The semantics of the language are consistent and intuitive. The language is strongly typed and block structured.

COSIM supports simple sequencing, conditional, and iteration operations. Control constructs are provided via "if," "loop," "while," and "skip" commands.

The COSIM compiler supports file inclusion and conditional compilation as well as cross-language calling interfaces for the SP to routines written in FORTRAN, C, Ada, and MC68040 Assembler.

Some of the COSIM language concepts described below are illustrated in Figure 10.

The basic building block of a COSIM program is called a *chore*. A chore is a unit of work to be executed on a particular processor. A specific chore always executes on the same processor.

All chores for a processor are collected into the *scheduling section* for that processor. The scheduling section for a processor defines the order in which chores are to be executed on that processor. The collection of all of the scheduling sections for the individual processors forms the overall *schedule* in a COSIM program. A COSIM program can have one and only one schedule.

Chores that are performed on parallel processors are called parallel chores. Parallel chores that are performed concurrently are referred to as an *activity*.

A particular chore may communicate and synchronize with a parallel chore over a unidirectional
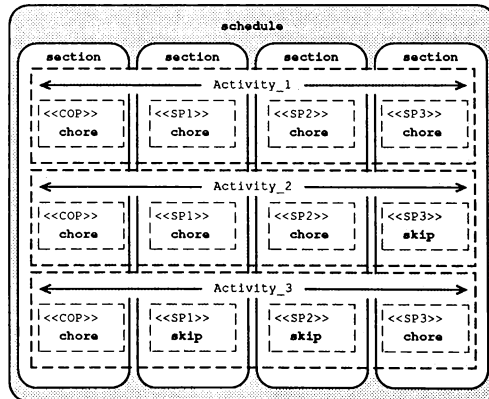
Figure 10. COSIM Language Concepts

*channel.** A channel connects a master processor (the COP in the AD RTS) with one or more slave processors. Channels allow synchronous, semisynchronous, and asychronous communications between chores on different processors.

Data to be communicated is contained in a structure called an *aggregate*. Data contained in a source aggregate is sent to a target aggregate over the channel. Aggregates are transmitted in optimized bursts at high data rates over the VMEbus in the AD RTS as part of the gather/scatter operations of the COP.

COSIM directly supports the following types of communications:

Synchronization. Synchronization of chores occurs and no data is exchanged between the processors.

Simple data exchange. The contents of a source aggregate are transmitted to a specified target aggregate.

Broadcast. The contents of a source aggregate are broadcast to a set of target aggregates on other processors.

Forked write. The contents of a source aggregate are partitioned into segments which are sent to target aggregates on other processors.

Forked read. The contents of two or more source aggregates are concatenated and sent to a target aggregate.

Figure 11 presents a simple example illustrating the flow of primetime events. This example involves obtaining input values for a simulation from external hardware via ADCs and providing outputs from the simulation through DACs. The looping operation indicated by the control flow line in Figure 11(A) provides the frame-rate control. This example shows two different PIRs being used, (each with an AIM unit) one for the analog input operations and the other for the analog output operations. (Note: In practice, these operations might be combined in one PIR; this would not substantially change the COSIM program.) The barrier synch function shown in Figure 11(A) is a

_____
* The basic channel concept was borrowed from another language called occam and extended for its use in COSIM.

programmed, once per frame synchronization of all devices controlled by the frame-rate timing mechanism for the simulation. Figure 11(B) shows the flow of primetime activities in the COSIM program for this example.

COSIM Run-Time Library. An extensive run-time library is provided as part of the COSIM package. These library routines provide a broad range of computational- and I/O-related functionality. Included in this library are routines for handling:

Operations on real and integer data;

COP and SP control;

Timer control;

Data conversion including packing, unpacking, and format conversion;

Bit-wise operations;

Message passing;

Ethernet communications support for the COP and SP;

Fiber-optic communications support for the COP;

VMEbus access for the COP; and

Support for the calibration, self-test, and on-line test capabilities incorporated in the various I/O devices.

COSIM Interact. The user interacts with the AD RTS through COSIM Interact. COSIM Interact is a software package that runs on the user's workstation and communicates with the AD RTS via the VIM. This software package provides a modern environment for monitoring and controlling all aspects of the distributed simulation handled within the AD RTS.

COSIM Interact allows the user to:

Allocate, reset, and control any specific AD RTS;

Load device-specific programs and data onto the various processors in an AD RTS;

Examine and modify various AD RTS registers and memory locations;

111

(A)

(B)

Figure 11. COSIM Example Illustrating Primetime Activities

Perform symbolic debugging including break-point management, single stepping, and variable manipulation;

Lock specified parameters to prevent these parameters from being changed accidentally;

Capture data for logging and displaying program variables;

Save and restore COSIM variables; and

Perform nonintrusive monitoring of variables.

Setting up and checking out all of the various I/O and interface channels and operations that are required as part of an HITL simulation can be a very tedious, time-consuming, and error-prone activity. The features of COSIM Interact, together with the AD RTS diagnostics, the low-level debugger, and the self-test and on-line test capabilities incorporated in the AD RTS I/O devices, greatly simplify the job of getting an HITL simulation set up and running correctly.

Compute Engine for the AD RTS

The MC68040 used in the SP was chosen primarily on the basis of I/O requirements with careful consideration given to cost/performance trade-offs. The SP is not restricted, though, to just I/O-related activities; it can also be used in a broad range of applications for handling some or all of the less demanding simulation model equations. FORTRAN and C cross-compilers are available for the SP to support these applications.

The SP is not intended to be the primary vehicle for handling the simulation model equations in very high performance (i.e., sub-millisecond frame-rate) applications. The AD RTS concept allows the use of a high-performance compute engine which can either be external to or incorporated in the AD RTS. As noted earlier, the external port on the COP provides one means for interfacing the AD RTS to a compute engine through either a fiber-optic or Ethernet link. Other ways of interfacing are also possible.

Section III Example Applications of the Multiframe I/O System

Using the AD RTS I/O System as an intelligent resource can off-load the main simulation processor of many of the I/O-related tasks that create a time burden on the system. Examples of these are floating-to-fixed and fixed-to-floating conversions, special signal conditioning, and bit packing and unpacking for discretes, to name only a few.

Figure 12 shows how data logging and record/playback can be delegated to an intelligent I/O resource.

The computed parameters necessary to initialize the simulated mission together with a complete I/O status block are recorded on a disk file with a mission time signature at a rate equal to the recall resolution requirement, e.g., playback that must be capable of initialization to any five-second interval within a two-hour mission recording.
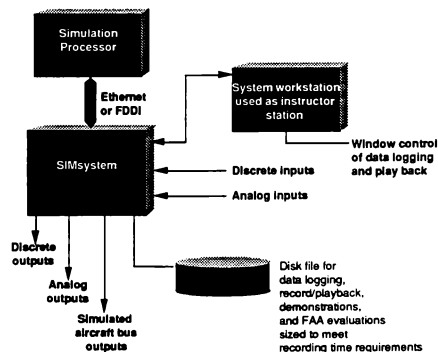
Figure 12   SIMsystem Data Logging Example

113

This then dictates that the full I/O status and all the long term integrands be stored in a file every five seconds and then only the input changes (δI) need to be stored at the normal I/O system rate, say 30 Hz, for the next five seconds.

The simulation processor's task is then simplified to only the loading of the standard initialization packet when the replay system is initialized and proceeding with its normal task. The I/O transfer to the host from the AD RTS will contain the stored inputs that allow the simulation to reproduce the recorded mission. The AD RTS can then do a compare every 5 seconds on the status of the replay calculations via the stored initialization file to monitor the status of the playback. If an actual on-board black box computer that requires a long term reset or realignment period is utilized in the simulation, the AD RTS can substitute stored data into its broadcast time slot on the simulated aircraft bus

until such a time as its output has aligned with the reset mission.

Figure 13(A) shows the typical approach used for current simulator connections with the extrapolation taking place in the visual system processor where the parameters necessary for proper compensation are often not available. Figure 13(B) shows the AD RTS being used to implement latency compensation techniques described earlier in this paper. The parameters needed for the additional compensation can be included in the standard data packet from the simulation processor and used in the floating-point format to condition the displacement vector outputs to the visual system. This gives the user the ability to experiment with the various library extrapolation routines or even to change routines on-line to meet special cases without disturbing the visual system software.
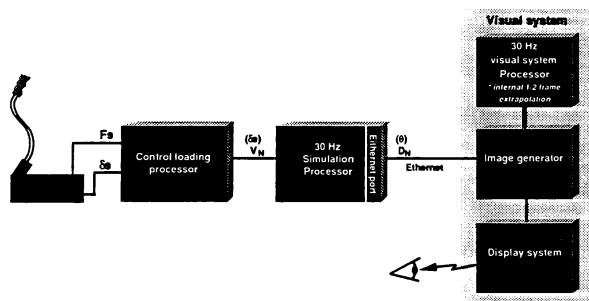
Current System (A)
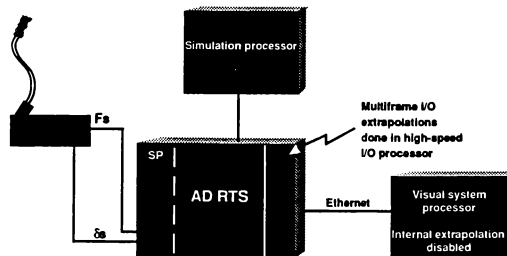
SIMsystem (B)

Figure 13 Latency Reduction Examples

References

1. Howe, R. M. "The Use of Mixed Integration Algorithms in State Space." *Transactions of the Society for Computer Simulation.* 7:1 (Mar 1990) 45-66.

2. Howe, R. M. "Dynamics of Digital Extrapolation and Interpolation." *Trans. of the Society for Computer Simulation.* 2:1 (Dec 1985), 169-87.

3. Howe, R. M. "Some Methods for Reducing the Effect of Time Delays in Flight Simulation." *Proc. of the AIAA Flight Simulation Technologies Conference.* Dayton, Ohio, Sept. 17-19, 1990, 225-232.

4. Applied Dynamics International. 1991. The SIMsystem: A New Product Line from Applied Dynamics International

APPLICATION OF EXISTING SIMULATION AND FLIGHT TEST RESULTS TO DEFINING 21ST CENTURY AIRCRAFT VERTICAL SITUATION DISPLAYS *

Moses Aronson **
Aronson Industries
3705 Wilder Lane
Orlando, FL 32804

Abstract

Future high speed aircraft such as the X-30A, National Aerospace Plane and the 21st Century combat and hypersonic commercial transports will not have a windshield as on current aircraft but will have the pilot and other flight operating personnel submerged in the fuselage. As a result, in most or all of the flight regimes, the pilot will be guided by vertical and horizontal situation displays on the cockpit instrument panel. Current cockpit instrumentation coupled with computerized flight control systems permit the pilot to perform limited maneuvers such as approach and landing, air intercepts, climbs and cruise. What if the computerized control system fails or is inadequate to perform the complex maneuvers, then the pilot must depend on these situation displays to perform his maneuvers manually.

Current electronic primary flight displays are not much larger than conventional instruments and are not precise enough for complex maneuvers. Essentially the subtended angle at the pilot's eye is too small and the resolution of the display too coarse.

Previous research on pilot control/ display performance in aircraft and in simulators with minimal time delay are examined to attempt to define adequate subtended angle and resolution requirements. Recommendations for cockpit designers based on this research are made and if the existing data is inadequate, additional research is identified. In addition, some hardware technologies are examined which can provide the defined subtended angle of view and resolution.

Introduction

With Chuck Yeager's breaking the "sonic wall" on October 14, 1947 in the X-1 airplane, the U.S. entered the supersonic age. However, the aircraft was equipped with instruments the size of and similar to those used in the first instrument flight on record achieved on September 24, 1929 by Lt. Jimmie Doolittle in a Consolidated NY-2 airplane. The X-1 canopy design required flush mounting to conform to the ogival surface of the airplane fuselage's nose. Here was the beginning of poor visibility for landing and take-off for supersonic aircraft. Since that time on, human factors and cockpit design engineers have struggled to find suitable means for

---

presenting the pilot with sufficient information from outside and inside the cockpit for him to perform his missions under direct control of his aircraft or at least under emergency conditions and control. Examination of new super or hypersonic aircraft designs show that there is a likelihood that the pilot will be submerged inside the fuselage without direct or with limited direct view of the outside world[1,2]. Hence his dependence on the primary vertical and horizontal flight displays in the cockpit. Which leads to the question of this paper, what kind of displays should the pilot have?

Statement of the problem

What kind of displays should the pilot have to answer the following questions:

1) Where in the world am I with respect to my end goal?
2) What is and what should be my velocity vector?
3) What is and what should be my attitude and/or angle of attack?
4) What should I do with the controls?[3]

The type of information depends on the mission phases. These phases were categorized in the Army-Navy Instrumentation Program (ANIP)[4] for military aviation, but can also be applied to commercial aviation as:

Take-off
Fly out
Closing (military only)
Engagement (military only)
Return
Landing.

For the purpose of this paper, only the Vertical Situation Display will be considered in answering the above questions for the various mission phases.

Analysis

The research to define the Vertical Situation Displays for supersonic flight began about 1946 at the University of Illinois under contract to the Special Devices Center of the Office of Naval Research.[5] This research addressed the problem of take-off and landing using pictorial vertical displays. Subsequent research under the ANIP starting in 1951 and the Joint Army and Navy Instrumentation Research (JANAIR) Project in 1960 addressed the improvement of piloted aircraft instrumentation systems, components and subsystems. Comparable programs in the USAF also started about 1955. The research results to be presented will address only the size and magnification of the Vertical Situation Display for pitch, roll, heading and flight path vector information. The question of size of the display has to do with both reading the display's indications and the

error in the control of the aircraft's flight path due to interpretation of the display's indications. The JANAIR project identified three types of forward looking Vertical Situation Displays used for flight control. They are nominally azimuth-elevation displays and portray aircraft pitch, roll, heading, angle of attack, glideslope error, steering error, etc. The display wherever possible will represent simultaneously the terminal goal, the path to the goal, aircraft attitude and flight path information and steering commands. The three types are: literal, analog and skeletal. As the terms describe, the first is essentially a picture of the real world; the second is an analog of the real world somewhat simplified, drawn by a digital computer, providing correct perspective of a three dimensional model and the dynamic response of the visual world while getting the input from sensors. The third is still pictorial but its content is minimal and made up of fragments similar to current head-up displays (HUD)

Literal displays

The sizes or subtended angle of view or field of view from the pilot's eye of literal displays and means for presenting the forward view have varied. Table I lists some of the systems evaluated. The results of those experiments may be summarized as follows: Optical periscope flights in References 5, 6, 7 found that a display field of view (FOV) of 30° x 30° with a magnification of 1.2 gave landing and take-off performance very close to that for contact flight. Reference 9 found that flying the traffic pattern in a DC-3 with an onboard TV camera system, prior to landing was difficult with all

pickup FOVs and the authors recommended that a display with a subtended angle of view of at least 45° with a magnification of 1.55 (or a 29° FOV for the pickup lens) was the minimum needed. Campbell, et al of Reference 8 found that the binocular 70° FOV in the B-17 periscope was inadequate to fly the then existing traffic pattern though adequate for landing. Though the view was binocular with a one power magnification, the apparent magnification was .8 if the scan was centered about the flight axis. Thus the system magnification to make things look normal was about 1.2. Subsequent to most of these tests (1948 - 1967), at least three studies addressed the differences in landing accuracy and approach technique, due to monocular or binocular vision. This consideration has some bearing on whether the proposed display should be monocular or binocular. Lewis and Krier[13] had shown a slight, but statistically non-significant, advantage in monocular landings. These were performed by NASA test pilots in a T-33 aircraft. The landing approaches were steeper for monocular vision and pilots reported a marked lack of confidence in their ability to judge height. Grosslight et al.,[14] reported that approaches and landings made by private pilots in a Beech Sport airplane showed no statistically reliable differences (p>.05) between the mean absolute error on monocular versus binocular landing touchdown point, averaged over all landings. It was also reported while monocular and binocular landings were similar with respect to accuracy, they were decidedly different with respect to approach altitude (monocular had higher and steeper flight path) and landings with monocular vision touched down beyond the target line while the binocular landing touchdown point tended to be short of the target line. The monocular

TABLE I. Literal Vertical Situation Displays

Reference	Aircraft	Display Input	Missions Evaluated	Input FOV deg. HxV	Display Means	Subtended Angle °HxV	Best Magnification	Variables
5	Cessna T-50	Periscope	C, L, T	15, 25, 35	Ground Glass Screen	30 x 30	1.2	Magnification, 6 Ss
6, 7	Cessna T-50 Aeronca	Periscope	"Oboe" Instrument Flt. Pattern	15, 25, 35	Ground Glass Screen	30 x 30	1.29	Magnification, 6 Ss
9	DC-3	CCTV in nose	S, L	11, 23, 48	TV Monitor 17" Diag.	17 x 12	1.55	Magnification, Ss
9	DC-3	Contact visual	S, L	Contact visual	4" Aperture in windscreen	18.5 - 21.9	—	Subjects
8	B-17	Periscope, Binocular	C, L	70 x 70	Eyepiece	70 x 70	1.2	Subjects
10	Remote HIMAT	CCTV	S, L	14 x 10	TV Monitor	28 x 21	—	3 Ss
11	T-33A	Contact visual	C, L	Contact visual	Mask on Canopy	5.7 x 30 to 140 x 74	—	FOV of Mask, 4 Ss
12	Cessna T-50	Periscope	S, L	10°40' x 11°50'	Ground Glass Screen	17 x 17	—	Restricted FOV - monocular, 6 Ss
12	Cessna T-50	Eyepiece & Vision directing shield	S, L	10°40' x 11°50'	Aperture in screen	17 x 17	—	Restricted FOV - binocular, 6 Ss

C = Circling Approach   L = Landing   Ss = Subjects; all pilots except as
S = Straight-in Approach   T = Take-off   noted

landings had a somewhat harder landing impact than the binocular ones. Roscoe[15] tries to explain why there would be differences between monoculary and binocular piloting and other activities!

All these displays provide the pilot with a contract flying presentation. Thus he still needs other instruments to give him absolute values of airspeed, altitude, rate of climb, rate of turn, heading, etc.

Contact analog displays

Contact analog displays are described by Carel[3] as the point perspective projection of a three dimensional model to a picture plane. The model contains reference objects significant for flight performance such as surface representing the local horizontal, usually called the ground plane, a surface representing the command path for the pilot to follow, usually called the flight path or "the highway in the sky" and other surfaces as objects useful during different phases of the mission. The hall mark of a contact analog is the display of surfaces whose kinematics are similar to those of real surfaces in the natural visual environment. In the microcosm of the panel mounted display where magnification may be other than unity, the displayed surfaces will still follow the laws of motion perspective and thus provide information coded in a fashion analogous to the coding provided in visual contact flight. The genesis of the contact analog display is the ANIP contract with General Electric in 1960. A subsequent contract was let to the Norden Division of United Aircraft Corp. in June 1964 for Universal Contact Analog Research. Between 1964 and 1965 both Bell Helicopter[16,17] and Norden Division[18] conducted evaluations of these displays in engineering simulators. The Bell evaluation was in a helicopter simulator on a motion platform while the Norden one was both fixed wing and VTOL in a fixed base simulator. It was not until 1983 that a contact analog display for fixed wing aircraft with command flight path was evaluated in flight using the Total in Flight Simulator (TIFS) aircraft under NADC sponsorship.[19] The display was on a 8" TV monitor. In 1985 another flight evaluation, this time using a Grumman F-14A at the Pacific Missile Test Center was conducted[20], though no results were published. The particulars of the above mentioned displays are summarized in Table II.

Though most of the contact analog experimentation was done in helicopters some of those results may be applicable to the fixed wing supersonic airplane. From Emery and Dougherty[17], where a comparison was made of pilot performance using the contact analog and conventional instruments, little difference was found in performance using either display system when the pilot is not stressed. When an additional side task was imposed on the pilot then performance with the contact analog remained relatively stable whereas performance with the standard instruments deteriorated with increasing load. The authors suggested that the superiority may be due to three factors:

1) Pilot may more quickly assimilate qualitative information from the pictorial display

2) Using conventional instruments the pilot samples one parameter of information per glance. With the pictorial display he accumulates information on more than one parameter per glance

3) Because of its size, the pictorial display permits use of peripheral vision.

The Norden results[18] for fixed wing aircraft flight under Basic Instrument Flight Maneuvers were as follows:

For straight-level flight, the experiments indicate that significantly lower errors resulted with the integrated display for all parameters except turn rate and bank angle. In the case of turn rate, pilot subjects reported that the response characteristics of the turn rate vector were much more instantaneous than for the standard turn needle. As a result, they tended to "chase" it from side to side with sudden control in an attempt to keep it nulled. In a 45° banked turn there were no significant differences between the contact analog and conventional instruments. In a transition to 1000 ft/min climb all but one of the parameters (vertical velocity) were performed significantly better on conventional instruments than on the advanced display. For take-off there were no significant differences while for ILS landing the integrated display was significantly better for reducing errors on glideslope and localizer. The NADC[19] results of flight tests in the TIFS aircraft with an approximate 9° x 9° FOV contact analog display showed a maximum deviation from the desired centerline of 212 feet for the contact analog compared with deviations as high as 4500

TABLE II. Contact Analog Displays

Reference	Display Manufacturer	Simulator	Missions Evaluated	Input FOV deg. HxV	Display Means	Display POV deg. HxV	Motion Platform	Variables
16, 17	Norden	Bell HU-1 Dynamic Motion	Helicopter Lift-off touchdown Hover & Cruise	Not Available	19" Diag. TV Monitor	28 x 22	6 DOF	Symbology Configuration, Contact Analog vs std. inst.
18	Norden Universal	SH-3A	Fixed wing Inst. Flt, Landing & Take-off	50 x 48.8	8" Diag. TV Monitor	12.2 x 9.2	Fixed Base	Contact Analog vs std. inst. 8 Ss
19	General Electric Intermetrics	TIFS aircraft	Turns, climb Landing	9.6 x 9.6 (est.)	8" Diag. TV Monitor	9.2 x 9.2	Flight	Contact Analog vs std. inst.

Ss = Subjects; all pilots
(est) = estimated

feet that occurred when conventional symbology was used. The figures represent an average. Some pilots flew with almost no deviation. Maintaining altitude caused more difficulty than maintaining track. Flights were made using a multiple way point track, with predetermined turns, climbs and descents, including simulated instruments approaches and accelerations and reductions in speed. One claim made for the system is that the work level remains constant and does not increase during approach.

While General Electric Co. did not manufacture a contact analog display for the cockpit, they applied this technology to providing Computer Generated Imagery (CGI) for visual simulation. Some indications for the subtended angle of the VDS needed by pilots to perform specific tasks can be obtained from simulator experiments which used visual field of view as a variable. Data was collected from simulator experiments where the simulator may have been mounted on a 6 degree of freedom (6 DOF) motion platform and had a system delay of 100 milliseconds or less. The main differences between the CGI visual simulation and the VSD mounted on the cockpit instrument panel is the requirement for the pilot to refocus his eyes and accommodate them on the instrument panel to read the instruments after viewing the projected visual scene at a distance of 10 - 20 feet from him. The summary of applicable experiments is shown in Table III. One of the earliest experiments using the Advanced Simulator for Pilot Training (ASPT)[21] showed that full FOV (300°H x 150°V) was consistently better when compared with the 48° x 36° limited FOV. This effect was particularly evident in the performance of the aileron roll and slow flight maneuvers which were roll sensitive. This indicated that peripheral vision cueing is extremely important for correctly maintaining level flight and bank

attitude. Tactical Air Command (TAC) pilots (A-10, F-4) using the ASPT[22] needed greater vertical coverage, about 12°, to properly perform refueling and take-off. While, the Strategic Air Command (SAC) pilots found their FOVs satisfactory. All pilots performed better with the full FOV. The TAC pilots using a 44° x 35° FOV required 270% more elapsed time to reach criterion than with the full FOV (essentially contact). The SAC pilots using the 48° x 38° FOV required 200% more elapsed time to reach criterion than with the full FOV. Collyer et al[23] reported that pilots using the narrow FOV performed carrier landings as well as with the large FOV visual simulation. Pilots using the narrow FOV performed the circling approach and landing using instruments and the visual display. On the Visual Technology Research Simulator (VTRS)[24] four visual system variables were explored, from instruments only to wide angle visual simulation. Pilots flying the narrow FOV had more difficulty maintaining altitude than those flying the Outside-in Display and the instruments only conditions, apparently because these later two had more precise indications of altitude. The wide FOV performance was more accurate in altitude than the narrow FOV. Lateral error was the same for narrow and wide FOV. The Outside-in and Flight Instrument conditions showed greater numbers of banks or large roll angles than the normal perspective displays because in these the change in bank angle was reflected by the roll of the relatively small aircraft image or the artificial horizon bar. A particular roll angle may have appeared large in the wide or narrow FOV displays but small in the Outside-in or the Flight Instrument Displays (160°H/48°H vs 2.35°H/8°). On the VTRS, in Reference 25, the wide FOV again had less roll variability than the narrow FOV during descent and at touchdown. The number of successful landings with either display for straight-in approaches was nearly the same.

TABLE III. Visual Simulation Experiments – Computer Generated Imagery

Reference	Simulator	Visual System	Missions Evaluated	Display Means	Display FOV deg. H x V	Motion Platform	Variables
21	ASPT(T-37A)	GE/Farrand	360° overhead, slow flight, Aileron Roll	Virtual image	360 x 150 48 x 36	6 DOF	FOV 3Ss
22	ASPT (A-10) (B-52) (F-4) (FB-111)	GE/Farrand- 1 window 1 window 3 windows 3 windows 7 windows	Rendezvous, Take-off, Landing, Aerial Refueling	Virtual image	44 x 35 48 x 38 115 x 49 143 x 46 300 x 150	no motion 6 DOF 6 DOF no motion both	6 TAC pilots 6 SAC pilots 6 TAC pilots 6 SAC pilots 12 Ss FOV
23	ASPT (A-10)	GE/Farrand	Carrier Landing, Circling or straight in	Virtual image	300 x 150 48 x 36	no motion	FOV 21 Ss
24	VTRS (T-2C)	Link/GE	Straight & Level Flt Aileron Roll	Projected on Dome	160 x 80 48 x 36 no visual outside-in	6 DOF	16 Ss non pilots FOV target 2.35°H backgnd 48° x 36°
25	VTRS (T-2C)	Link/GE	Carrier Landing, Circling & straight in	Projected on Dome	160 x 80 48 x 36	6 DOF no motion	8 Ss FOV

Ss = Subjects; all pilots exept as noted

Skeletal vertical situation displays

Carel in Reference 3 proposed certain skeletal displays but none have been developed. What is now available are head up displays (HUD) which present skeletal information. The first HUDs were developed for low visibility landings. Now they include air to air combat and air to surface attack information. Two computational bases are used: Flight path trajectory and attitude. The flight path trajectory shows by means of a symbol the position of the flight path in space or for landing where the impact will be with the present control and power settings when superimposed on the out the cockpit scene in front of the aircraft. Variation of the flight path symbology is the Klopfstein imagery which provides a drawing of a typical runway with the flight path marker superimposed on the outside world runway and does not need to furnish any numerical readouts.[26] The attitude basis is an extension of the present attitude indicator. In view of the lack of interest in installing HUDs in commercial aircraft there are only subjective pilot evaluations of their performance. The principal proponent of the trajectory flight path indication is Gilbert Klopfstein of the French Air Force.[26] The DC-9-80 HUD with flight path trajectory[27] was considered adequate for go around on missed approaches. As stressed by pilots and the HUD manufacturers these are supplements to Category 3A and 3B landings where the pilot still must have final visual contact with the runway. The major advantage cited was that all aircraft information pertaining to heading, glideslope, height, speed, etc., is presented on the HUD thus eliminating the time factor of 2 - 3 seconds pilots need to transition physically and psychologically from head down instrument flying to head up visual flying at a critical point in the landing process.

Magnification and relation to "g" forces

In addition to reviewing Vertical Situation Displays with regard to FOV, the question of magnification, that is, the image feature's size in relation to the outside the cockpit visual scene feature's size must be addressed. Carel[3] pointed out that if the synthetic outside-the-cockpit scene is compressed (neglecting imposition of real outside world visual input on the same screen), the pilot's impression of the ride of the vehicle is smooth (that is small changes in attitude are not noticed by the pilot and not corrected). If the synthetic scene is magnified, then the pilot will feel the ride of the vehicle to be bumpy as a result of "chasing" the horizon bar. This has occurred in actual flight. Weiner[28] describes that on transition from conventional instruments (3" diameter attitude indicator) to the electronic attitude indicator (EAI 6" CRT) and from pitch graduation of 5° on the 3" indicator to 2.5° on the EAI, pilots during the first 200 hours after introduction to this display in the DC-9-80 found that the magnified scale created an impression that their flying was rougher in holding a constant attitude angle. A similar experience is described in Reference 29 in flying with the Flight Dynamics HUD in a Boeing 727, where "a one degree pitch change would be hardly noticed on a head down attitude indicator, but a one degree pitch change when seen thru the HUD appears as a major attitude divergence. This was equally true for a one degree change in heading when seen first on a head down compass card and then thru the HUD. Because of this, one has a tendency initially to 'chase' the symbology in the HUD because attitude deviations appear to be so much larger than they really are. These reactions subside relatively

quickly as (the pilot) becomes accustomed to the sensitivity of the display." In an argument in the opposite direction Walters[30] reported that RAE pilots opposed 1:1 artificial horizon movement for highly maneuverable aircraft, and preferred a scaling of .2 to reduce the activity of the pitch bar or own plane reference in the HUD.

Pilot interpretation of displays

Work carried out by the Human Engineering Division of the USAF Aerospace Medical Research Laboratory[31] reported that one problem to solve is to find ways by which the real and simulated images do not get scrambled together like a double exposure. Ways of avoiding this problem are just beginning to be examined in the AF Super-cockpit program. Test results show that the symbols such as stick figures can be superimposed on real world images but not real images such as grey scale images. For low fast operations at night the pilot cannot use IR enlarged views because these are at variance with the visual cues presented by TV or periscope. If the pilot loses confidence in what the display presents he tends to fly higher and slower. The pilots get much of their ground avoidance information from the perceived angular movement of terrain features or objects on the ground. Another problem was integrating the night IR television picture with the calligraphic symbology in a wide FOV HUD to improve the pilot's ability to navigate, locate targets off his flight path and maneuver, either to avoid defenses or to attack targets. The maneuver capability requirement dictated that the FOV be large both vertically and horizontally since the aircraft bank angles increase, the need for the pilot to have a large vertical FOV- in effect to see up the VSD- increases rapidly.[32]

Results

Considering all the data presented it is possible to summarize a set of suggested recommendations for VSDs. Some will apply to hypersonic transports at speeds of $M = 2 - 5$ and others to supersonic/ hypersonic fighter aircraft. These are presented in Table IV. Each mission's requirements are listed separately. The magnification and sensor recommendations are distilled from the previous data while the data rate recommendations are from Carel[3]. The terrain following display recommendations come form preliminary studies of Reference 33. Where specific terrain or aerial objects must be identified the sensor should be TV or thermal imagery. For the aileron rolls or straight and level flight missions contact analog or computer generated imagery is recommended. The FOV for aerial combat is based on personal computations taking into account higher maneuvering speeds and greater line of sight angles at higher"g"s than current fighter aircraft. The FOV given for Weapons delivery is a compromise based on data from at least six experiments using simulators and varying air-to-ground tasks. The display size is based on the 28" viewing distance given in Military Standard for Human Engineering Design Criteria for Military Systems, Equipment and Facilities: MIL-STD-1472. To reduce all display requirements of Table IV to one size for inclusion in the airplane instrument panel, using the largest dimensions listed, then the display dimensions are 25" horizontal by 18" vertical or about 450 square inches. This size may exceed the instrument panel portion currently assigned to either the pilot or copilot. If the cockpit is submerged it may be located in a wider portion of the fuselage and this

TABLE IV. Suggested Requirements for Vertical Situation Displays

Mission	Sensor	Sensor/CGI FOV deg. H x V	Magnification	Display FOV deg. H x V	Size of Face in. H x V	Data Rate per sec.
Aerial Refueling	TV	115 x 49	.33	38 x 16	19 x 8	60
Carrier Landing	TV	48 x 36	1.0	48 x 36	25 x 18	60
Circling Approach & Landing on Runway	TV	48 x 36	1.0	48 x 36	25 x 18	60
Take-off	TV	30 x 20	1.0	30 x 20	15 x 10	30
Straight & Level Flight	CGI	48 x 36	1.0	48 x 36	25 x 18	30
Aileron Rolls	CGI	48 x 36	1.0	48 x 36	25 x 18	60
Terrain Following	TV/Thermal imagery	48 x 36/ 24 x 18	1.0 2.0	48 x 36	25 x 18	30
Aerial Combat	TV	47 x 42	.5	23.5 x 21	11.6 x 10.4	60
Weapons Delivery	TV	90 x 100	.5	45 x 50	23.2 x 26.1	60

size could be accommodated. This is contrasted with the AF Avionics Laboratory program at McDonnel Douglas which forecasts a big picture display of 300 square inches for the cockpit and it could include more than just attitude and flight path information.[34] The implications of this size and possible means for providing this display are covered next.

Discussion

How will a display 25" x 18" in size be provided? In ground based engineering simulators such as those at Boeing, McDonnel Douglas and Northrup, this is provided by a rear projection TV system since the nose of the proposed aircraft does not have to be modelled. For aircraft use the likeliest prospects are flat panel displays. Forecasts are that Thin Film Electroluminescent (TFEL) and Liquid Crystal (LC) panels will mature from the current 4" x 8" panels with a contrast ratio of 20:1 and a resolution and a resolution of 64 lines per inch to 15" x 30" panels in 5 to 15 years. Carel[3] recommended that the required resolution should be about 2400 TV lines in azimuth and 2000 TV lines in elevation or about twice the currently achieved. He points out that if the desired resolution cannot be achieved, then, in a trade-off, make the elevation resolution something like twice the azimuth because for forward looking displays range loss due to foreshortening compression on the elevation axis takes place very rapidly.

Another factor to consider is that the pilots will accept a literal or contact analog display quicker than conventional instruments or symbology because it is easier to interpret and provides unambiguous information. The pilot can respond immediately without having to mentally transform data from numbers to a visual picture. Based on pilot comments it appears that the literal or contact analog type of VSD provides reduced workload for pilots during critical mission segments such as approach and landing, and weapons delivery. This also raises the question as to whether the display should provide binocular or monocular vision. Earlier tests have not really shown a preference for binocular vision, hence a monocular display has been suggested. But this approach may change as more experiments are being performed to answer this question.

A recent experiment by Rockwell, applicable to the NASP, indicated that pilots in a TF104G simulating the NASP, preferred a stereoscopic view of the runway by means of a wide angle mirror periscope over a synthetic vision system for approach, touchdown and rollout.[35] NASA Langley is exploring the use of 3-D stereo flight displays using collimated optics, synthetic scenes with pilots wearing 3-D LCD stereo glasses.[36] Boeing under contract to Wright Aeronautical Laboratories is designing simulation equipment to "evaluate the potential payoff in situational awareness of adding retinal disparity (3-D) to picture formats for use in advanced aircraft and to advance the evolution of the pictorial formats themselves.[37] Also Evans and Sutherland is exploring techniques for presenting stereoscopic scenes in flight simulators.[38]

While Table IV shows certain recommended magnification values for each mission, the magnitude of magnification has not been resolved either. Despite Roscoe's research[5,6,7 & 12] showing an average magnification of 1.21, Meehan and Triggs' research[39] shows an average magnification of 1.03 to 1.11 and a difference between monocular and binocular vision required magnification. They conclude that visual accommodation plays an important role in judgments of size and that appropriate occulometer adjustments may contribute to a reduction in perceived size with imaging displays. Though not cited in the Tables, the pilot's eye distance to the image display surface differed for all the experimental data. Thus, eye accommodation, stereoscopis, dark focus and possibly other physiological effects in the vision channel may modify these recommendations.

The question of adequate peripheral vision for control of banks and rolls with the suggested display size has not been completely answered. Some experiments showed that pilots could maintain lateral control in a tracking task within 1 - 2° with a 48° FOV.

A replacement for tabulated operational manuals data or doctrine standard values is the "command flight path in the sky". Here the pilot attempts to either maintain his position on a "highway in the sky" which defines his future path or flies

"wing man" to another airplane image in the display which is programmed to fly the desired or command path. The inputs for the command path are the onboard sensors and computers which will be programmed by the pilot before take-off. A simpler alternate to the "flight path in the sky" is the flight path or command velocity vector symbol which the pilot will "fly to" (make his current flight path vector coincide with the command symbol). The sensors and data computations for deriving the information for the command flight path vector would be the same as for the "flight path in the sky". In both cases a determination must be made of how much deviation from the command flight profile is acceptable.

Conclusion

This paper has examined the size for a VSD to be installed in a very high speed airplane which has a submerged cockpit. The VSD is a substitute for the pilot's direct out-the-cockpit view. The suggested display size is 25" wide and 18" high. It will probably be a flat panel display such as TFEL or LC. The desired display content and its magnification will be selected by the pilot for each particular mission segment. This display can have a "flight path in the sky" or a command velocity vector added to it to provide the pilot with desired or optimum flight trajectory for each mission segment. Additional research has also been identified to meet the 21st Century airplane control goals.

References

1. Nordwall, Bruce D., "Advanced Cockpit Development Effort Signals Wide Industry Involvement," Aviation Week & Space Technology, p. 72 (Apr 20, 1987).

2. Ott, James, "Contractors Urge Further Research to Validate HSCT Technologies," Aviation Week & Space Technology, p. 40 (Oct 17, 1988).

3. Carel, Walter L., "Pictorial Displays for Flight," Hughes Aircraft Co. Technical Report 2732.01/40 (Dec 1965).

4. Dunlap and Associates, "An Examination of Pilot Information Requirements," Prepared for Douglas Aircraft Co., Long Beach, CA (AD 401662) (Nov 30, 1962).

5. Roscoe, Stanley N., Hasler, Scott G., and Dougherty, Dora J., "Flight by Periscope: Making Take offs and Landings; The Influence of Image Magnification, Practice, and Various Conditions of Flight," Human Factors, p. 13 (Feb 1966).

6. Roscoe, Stanley N., "Flight by Periscope: I., Perform-ing An Instrument Flight Pattern., The Influence of Screen Size and Image Magnification," Aeronautical Bulletin No. 9, University of Illinois Institute of Aviation, Urbana, IL (Mar 1951).

7. Roscoe, S.N. and Williams, Jr., A.C., "Pilot Perfor-mance in Instrument Flight as a Function of the Extent of and Distribution of Visible Horizon," Technical Report SDC 71-6-3, Port Washington, NY (June 1949).

8. Campbell, C.J., McEachern, L.J., and Marg. E., "Flight by Periscope," WADC Technical Report 55-142 Aero Medical Laboratory, WADC, WPAFB, Ohio (Mar 1955).

9. Kibort, B.R. and Drinkwater, F.J., "A Flight Study of Manual Blind Landing Performance Using Closed Circuit Television Displays," NASA TN-D-2252, Ames Research Center, Moffett Field, CA (May 1964).

10. Sarrafian, Shahan K., "Simulator Evaluation of a Remotely Piloted Vehicle Lateral Landing Task Using a Visual Display," in AIAA Atmospheric Flight Mechanics Conference, A Collection of Technical Papers, Seattle, WA (Aug 21 - 23, 1984).

11. Perry, John J., Dana, William H., and Bacon, Donald Jr., C., "Flight Investigation of the Landing Task in a Jet Trainer with Restricted Fields of View," NASA TN D-4018, Flight Research Center, Edwards, CA (July 1967).

12. Roscoe, Stanley N., "The Effects of Eliminating Binocular & Peripheral Monocular Visual Cues Upon Airplane Pilot Performance in Landing," Journal of Applied Psychology, Vol. 32, No. 6, pp. 649 - 662 (Dec 1948).

13. Lewis, Charles E. Jr. and Krier, Gary E., "Flight Research Program: XIV. Landing Performance in Jet Aircraft after Loss of Binocular Vision." Aerospace Medicine, 40, pp. 957 - 963 (1969).

14. Grosslight, J.A., Fletcher, Harold J., Masterton, R. Bruce, and Hagen, Richard, "Monocular Vision and Landing Performance in General Aviation Pilots: Cyclops Revisited," Human Factors, 20(1) pp. 27 - 33 (1978).

15. Roscoe, Stanley N., "Judgments of Size and Distance with Imaging Displays," Human Factors, pp. 617 - 629 (Dec 1984).

16. Emery, J.H. and Dougherty, D., "Contact Analog Simulator Evaluations: Climb out, Low Cruise, and Descent Maneuvers,: Nonr 1076(00) Technical report D288-421-017, Bell Helicopter Co., Fort Worth, TX (May 1964).

17. Dougherty, Dora J., Emery, J.H., and Curtin, J.G., "Comparison of Perceptual Work Load in Flying Standard Instrumentation and the Contact Analog Vertical Display," Nonr 1076(00) Technical Report D228-421-019, Bell Helicop-ter Co., Fort Worth, TX (Dec 1964).

18. Williams, Peter R. and Kronholm, Marshall B., "Technical Report on Simulations Studies of an Integrated Electronic Vertical Display," Nonr 4489(00) Report No. 1161-R-0021, Norden Div., United Aircraft Corp., Norwalk, CN (Dec 1965).

19. Stein, Kenneth J., "Navy Evaluates Pictorial Cockpit Display," Aviation Week & Space Technology, p. 83 (Sep 12, 1983).

20. Elson, Benjaming M., "Planners Optimize Pilot/ Cockpit Interface in High Performance Aircraft," Aviation Week & Space Technology, p. 257 (Mar 18, 1985).

21. Waters, B.K. Grunzke, P.M., Irish, P.A., and Fuller, J.H. Jr., "Preliminary Investigation of Motion, Visual and G-Seat Effects in the Advanced Simulator for Undergraduate Pilot Training (ASUPT)," in Proceedings of the AIAA Visual

& Motion Simulation Conference, Dayton, OH (Apr 26 - 28, 1976).

22. Woodruff., R.R., Longridge, T.M., Irish, P.A., and Jeffreys, R.T., "Pilot Performance in Simulated Aerial Refueling as a Function of Tanker Model Complexity and Visual Display Field-of-View," AFHRL TR 78 - 98, Brooks AFB, TX (May 1979).

23. Collyer, S.C., Ricard, G.C., Anderson, M., Westra, D.P., and Perry, R.A., "Field of View Requirements for Carrier Landing Training," NAVTRAEQUIPCEN IH-319/AFHRLTR 80-10, Orlando FL (AD-A087012) (June 1980).

24. Hennessy, R.T., Lintern, G., and Collyer, S.C., "Unconventional Visual Displays for Flight Training," Technical Report NAVTRAEQUIPCEN 78-C-0060-5, Orlando, FL (Nov 1981).

25. Westra, D.P., Simon, C.W., Collyer, S.C., and Chambers, W.C., "Simulator Design Features for Carrier Landings: I. Performance Experiments," Technical Report NAVTRAEQUIPCEN 78-C-0060-7, Orlando, FL (Dec 1981).

26. Ropelewski, R.R., "Head-up Display Systems Evaluated," Aviation Week & Space Technology, P. 70 (Jan 10, 1977).

27. Smith, B.A., "Super 80 HUD has Immersed Optics," Aviation Week & Space Technology, P. 61 (Aug 28, 1978).

28. Weiner, E.L., "Human Factors in Cockpit Automation: A Field Study on Flight Crew Transition," NASA CR 177333 (July 1985).

29. Ropelewski, R.R., "FAA Nears Transport HUD Certification," Aviation Week & Space Technology, p. 145 (Oct 1, 1984).

30. Walters, D.J., "The Electronic Display of Flight Data," in Problems of the Cockpit Environment, AGARD

EXTENDED SUMMARIES 4, (AD 685268), p. 29-1 (Nov 1968).

31. Lerner, E.J., "Helping the Pilot Handle the Supercockpit," Aerospace America, p. 28 (Feb 1987).

32. _____,"F-16's Head up Display Developed for Lantirn," Aviation Week & Space Technology, p. 141 (June 8, 1981).

33. Zacharias, G.L., Caglayan, A.K., and Sinacori, J.B., "Visual Cueing Model for Terrain-Following Applications," in a Collection of Technical Papers of AIAA Flight Simulation Technologies Conference, (AIAA No. 83-1081), Niagara Falls, NY (June 13 - 15, 1983).

34. Stein, Kenneth J., "Complementary Displays Could Provide Panoramic, Detailed Battle Area Views," Aviation Week & Space Technology, p. 111 (Dec 1, 1986).

35. Scott, W.B.: "Rockwell's Simulator Emulates NASP Flight Characteristics." Aviation Week & Space Technology p. 50 (Oct 23, 1989).

36. Phillips, E.H.: "Experiments Show Collimated Optics Increase Depth of 3-D Flight Displays", Aviation Week & Space Technology p. 56 (Sept 17, 1990).

37. Ways, T.C.; Hobbs, R.E.; Quacy-White, J. & Gilmour, J.D.: "3-D Imagery Cockpit Display Development." WRDC TR 90-7003 (Aug 1990) (AD-A226411).

38. Tidwell, R.P.: "Stereopsis as a Visual Cue in Flight Simulation", Journal of Aircraft, 27(8) p. 731 - 732 (Aug 1990).

39. Meehan, J.W. and Triggs, T.J.: "Magnification Effects with Imaging Displays Depend on Scene Content and Viewing Conditions." Human Factors 30(4) p. 487 – 494 (1988).

EFFECTS OF TIME DELAYS ON HEAD TRACKING PERFORMANCE
AND THE BENEFITS OF LAG COMPENSATION BY IMAGE DEFLECTION

Richard H Y So[*]
Michael J Griffin
Human Factors Research Unit
Institute of Sound and Vibration Research
University of Southampton, Southampton
SO9 5NH, England

Abstract

Images on head-coupled systems are delayed by latencies in measuring head position and generating computer graphics. The objectives of this study were: (i) investigate the effects of time delays on head tracking performance, (ii) evaluate the use of an image deflection technique to reduce deleterious effects of delayed images and (iii) investigate the application of a head position prediction algorithm to enhance the benefits of image deflection. There were significant decreases in head tracking performance when lags of 40ms or more were added to a system with an inherent 40ms lag. Lag compensation by image deflection significantly improved tracking peformance with lags up to 380ms. However, by deflecting the delayed image back to its pre-lag angular position, part of the picture was displaced beyond the edge of the screen. The amount of deflection required was reduced by a simple means of predicting the position of the head before applying deflection. Improved means of predicting head position would further reduce the required image deflection.

Introduction

Head-coupled systems

Head-coupled systems were developed to improve the visual interface between human operators and aircraft flight control, navigation and weapon systems[1]. The helmet-mounted display consists of an image source which can be mounted on a flying helmet. The image is optically superimposed on the operator's view of the outside world regardless of the head orientation. The helmet-pointing system is a device which measures angular orientation of the head relative to a fixed reference (see Figure 1).

There are various potential uses of head-coupled systems. When a helmet-mounted display and a helmet-pointing system are used together, the operator can designate a target by moving the head so as to place the target behind an aiming reticle presented on the display. Target cueing and other information may also be presented on the helmet-mounted display.

Head-coupled systems can also be used to present views from a camera whose orientation is slaved to the helmet-pointing system. As the head is turned the direction of the camera follows the head pointing angle.

―――――――――――
[*]Member of AIAA
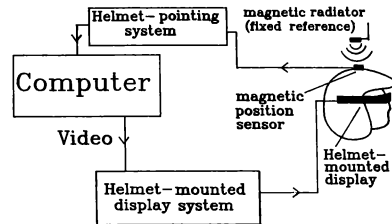
Figure 1   Block diagram of the experimental head-coupled system.

Computer-generated images can be presented on helmet-mounted displays to simulate a view of the real world for either research, training or inflight assistance. This approach can be extended to present a 'virtual cockpit' to the pilot with virtual displays and even multi-function virtual switches[2].

The problem

When a helmet-pointing system is used to control the location of images on a helmet-mounted display, any delay between the moment at which head position is sampled and the moment at which the corresponding image is presented will result in the image 'floating' on the screen. This is subjectively disturbing and may result in degraded performance. For example, a space stationary object may appear to "swim" during the start and finish of head motion[3]. Woodruff et al. reported that during an investigation of helmet-mounted displays for flight simulation (a head-coupled simulator), more than one-third of the pilots who attempted the task were unable to obtain satisfactory results and this was considered to indicate not inferior flying but failure to adapt to the inherent delay in the simulation system[4].

In a head-coupled simulator, the computation delays involved in providing an updated computer-generated scene are dependent on the complexity of the image. When a forward-looking-infra-red (FLIR) camera is slaved to a helmet-pointing system the delays depend on the inertia of the system, and the camera may lag behind the line-of-sight by up to one second.

Solutions

There are three possible solutions to the problems caused by delays: reduction of the time delay at source, deflection of the image to the

correct position, or head position prediction to compensate the time delays.

Head position prediction may be useful to reduce large position errors due to time delays but image deflection may be needed to remove remaining errors.

Reduction at source   With the advance in computer technologies, there is potential for reducing the processing time involved in graphics generation. However, this solution requires money and could be bulky.   This solution cannot be applied to a system with moving mechanical parts, such as a head-slaved camera.

Image Deflection   An alternative solution is to deflect the delayed image to the currently correct angular position so that the image maintains correct correspondence with the outside world. Image deflection on helmet-mounted displays was originally developed in the Human Factors Research Unit at the Institute of Sound and Vibration Research for the stabilisation of images on helmet-mounted displays exposed to vibration[5]. Applied to a head-slaved camera, the instantaneous angular separation between the camera and the line-of-sight is measured and translated into horizontal and vertical offsets, which are then used to deflect the video image on the helmet-mounted display. Applied to a computational delay, the offset can be measured and used by the graphics generation software to displace the image.

The principle of the image deflection technique is explained in the Appendix. The position error at any time is proportional to both the time delay and the head velocity. If these are large, the image may need to be deflected beyond the field-of-view; performance would then deteriorate rapidly. This analysis is consistent with results of unpublished studies previously conducted at the Institute of Sound and Vibration Research. Image deflection is therefore only applicable to the compensation of small position errors.

Head position prediction   This option utilises a signal processing algorithm to predict the future head position based on the past head movement time history. For random movements of the head, prediction is, by definition, impossible. However, due to the limited tracking bandwidth of the human head (0 to 1.5 Hz), prediction of head position becomes possible. The use of head position prediction to compensate for time delays has been previously investigated.   List reported a simulation study of the use of a simple non-linear prediction algorithm to compensate for time delays occurring during high velocity step movements of the head[5]. The algorithm used acceleration data and was reported to have been successfully implemented in a fiber optic helmet-mounted display.   Albrecht utilized an adaptive least-mean-square predictor to predict pilot head look direction[7].   Simulations showed that the predictor was capable of good predictions for input signals that change their characteristics linearly with time (e.g. a swept sine with decreasing amplitude) but needed improvement to predict head movements whose characteristics change randomly with time.

Head position prediction may bring the calculated head position close to the instantaneous position. However, due to noise and the randomness of the measured head movements, errors remain.

Objectives

The experiment reported in this paper investigated a lag compensation technique combining both head position prediction and image deflection. A block diagram illustrating the two methods is shown in Figure 2.   Head position prediction was performed first to reduce the difference between the instantaneous head position and the delayed head position.   Image deflection was then used to eliminate any remaining difference.
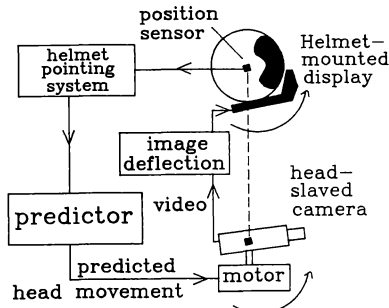
Figure 2   Diagramatic illustration of the lag compensation method involving combined head position prediction and image deflection.

An experiment was conducted to determine the effects of time delays on head tracking performance and assess the use of a simple head position prediction algorithm to enhance the benefits obtained by image deflection. Measurements of head tracking performance were made with various time delays and combinations of image deflection and head prediction.   The hypotheses were: i) head tracking performance would be degraded by time delays between head movement and image movement; ii) image deflection would reduce the performance degradation due to time delays but introduce a restriction on the field-of-view; iii) a simple prediction algorithm would enhance the benefits of image deflection by reducing the restriction on the field-of-view.

Materials and Methods

Apparatus

The head-coupled system used in this study consisted of magnetic coils mounted on top of a helmet, to detect helmet movement, and a miniature cathode ray tube mounted to the side of a helmet to

present a virtual image 17 by 17 degrees through a collimating optical arrangement. The configuration of the experimental apparatus is summarised in Figure 3. The helmet-pointing system was a Ferranti SPASYN type 101 and the helmet-mounted display was a Hughes monocular display model SD/HMD-001. A host computer was employed to control the target presentation and data acquisition. Image deflection and head position prediction were also performed within this computer (see Table 1 for prediction algorithm). The minimum system time delay was 40ms.

An open-cross reticle was presented at the centre of the display while the target was represented by a circle[8]. The position of the circle was derived from the absolute position of the target and the present (or delayed) line-of-sight measured by helmet-pointing system.

Method and Design

Subjects wearing the experimental head-coupled system were asked to move their heads to track a circular target viewed at optical infinity on the helmet-mounted display. The target motions were Gaussian random functions integrated twice and band-passed between 0.01 Hz and 0.63 Hz. The same target motions were used in both axes (vertical and horizontal), except one was presented in reverse order. Each tracking task lasted for 60 seconds. An x-y plot of the target forcing functions, the probability distribution of target radial velocity, and target movement power spectral density are shown in Figures 4 to 6.

Figure 3   Arrangement of the subject with helmet-mounted display and helmet-pointing sensor.
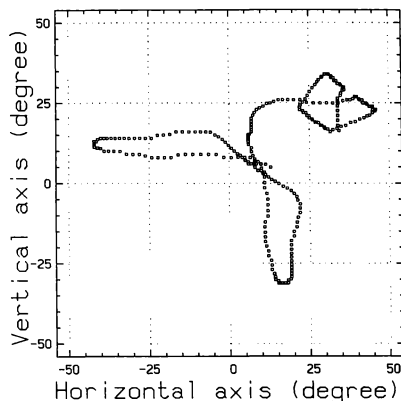
Figure 4   A typical target motion (each dot separated by 0.2 seconds).

Table 1   The simple extrapolation algorithm used to predict the head position based on constant velocity assumption

$$\hat{H}(t+n) \quad = \quad H(t) + (\text{head velocity}) \times (n)$$

where

$$\text{head velocity (instantaneous)} \quad = \quad \frac{H(t) - H(t-4t_s)}{4t_s}$$

Key:

$H(t)$ = measured head position at time t ms

n = prediction time (ms)

$\hat{H}(t+n)$ = predicted head position at $(t + n)$ ms

t_s = 20 ms (sampling period)
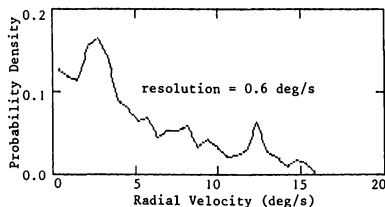
Figure 5   Probability distribution of target radial velocity.
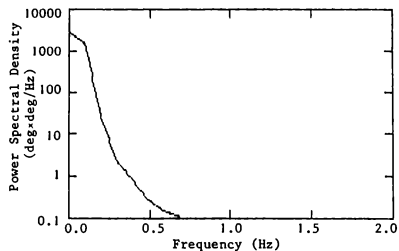
Figure 6   Power spectral density of the target forcing function angular displacement (same for both axes; 0.097 Hz resolution, 48 degrees of freedom).

Twelve subjects repeated the task with eight time delays (0, 40, 100, 160, 220, 280, 340, 380ms) and with three lag compensating conditions (no compensation, image deflection, head position prediction with image deflection). The sequence of presentations was balanced using a random block design. Twelve similar target motions were used to minimize learning of target motions. At the end of each tracking task subjects were asked to estimate the degree of task difficulty on a six point scale (Table 2).

Table 2 Six-point scale of difficulty

0	=	not difficult
1	=	a little difficult
2	=	fairly difficult
3	=	difficult
4	=	very difficult
5	=	extremely difficult

Written instructions were given to the subjects prior to the experiment. Four practice tracking tasks each lasting for 30 seconds were then presented to ensure that the subjects were familiarized with the experimental head-coupled system. (Previous studies have found that little learning is required during head tracking[6].) Throughout the experiment, subjects were screened within a darkened environment to eliminate visual information other than that perceived by the right eye through the helmet-mounted display.

<u>Results</u>

Head tracking performance measured in terms of radial error, percentage time-on-target (calculated with a reticle size of 1.5 degree diameter) and subjective difficulty rating are shown in Figures 7 to 9.

Friedman two-way analyses of variance were performed to determine effects of time delays within the three lag compensation conditions. Results showed that time delays had a significant effect on tracking performance for the condition where no lag compensation was applied ($p<0.001$).
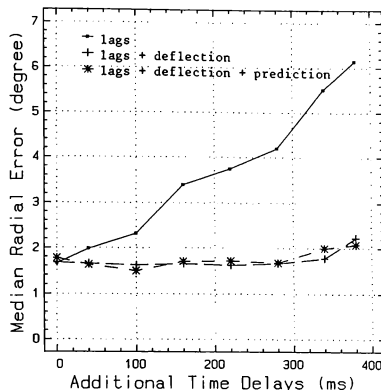
Figure 7   Radial error for different time delays (median of 12 subjects).
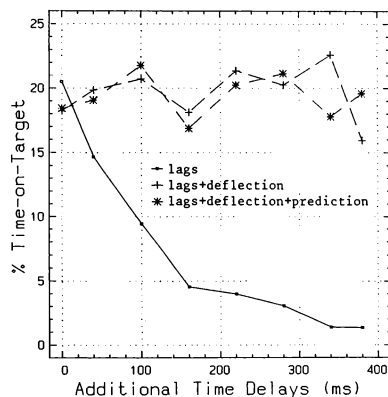
Figure 8   Percentage time-on-target for different time delays (median of 12 subjects).
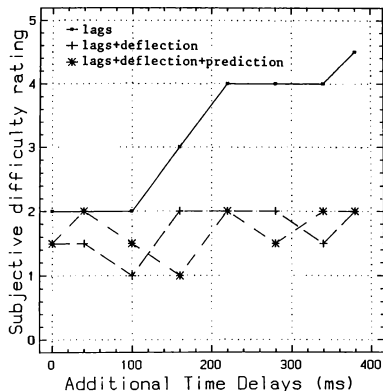
Figure 9   Subjective difficulty rating for different time delays (median of 12 subjects).

Figure 11   Root-mean-square values of image deflection used to compensate for different time delays (vertical axis, median of 12 subjects).

To assess the benefits of head position prediction, the amount of image deflection used in both axes was recorded during the conditions where image deflection was applied with and without head position prediction. Results in both axes are shown in Figures 10 and 11. Friedman two-way analyses of variance showed that for lags less than 280ms the deflection required was reduced significantly by head position prediction ($p<0.01$).

Figure 10   Root-mean-square values of image deflection used to compensate for different time delays (horizontal axis, median of 12 subjects).

Discussion

Effects of lags

For all objective measures of head tracking performance there was a significant degradation with lags greater than, or equal to, 100ms ($p<0.01$, excluding the system lag of 40ms). With the percentage time-on-target measurements, the threshold delay for significant degradation in tracking performance was 40ms ($p<0.05$, excluding the system lag of 40ms). Since complex computer generated graphics scenes presented on helmet-mounted displays can suffer lags of the order of 80 to 100ms[3,9], the finding of such a short threshold lag has important implications on the future development of head-coupled systems.

Effects of lag compensation by image deflection

Inspection of Figures 7 to 9 reveals that the image deflection technique (both with and without head position prediction) significantly reduced radial error, increased percentage time-on-target and improved the subjective difficulty rating. This is confirmed by the results of the Friedman two-way analysis of variance by ranks: there was no significant degradation in performance for any lag condition when using image deflection. This confirms that the image deflection technique totally restored the image to the correct position. Tracking performance was the same as without any lag despite the restricted field-of-view. The lack of an effect of the restriction on the field-of-view may be because once the target was captured within sight it could be kept around the centre part of the total field-of-view. The absence of any rapid search movements also meant that the reduction in the field-of-view was not large.

The perfect lag compensation by image deflection used here relied on two assumptions: (i) a knowledge of the total system time delay (ii) no distortion in the image from deflection in the vertical and horizontal axes. The former assumption will often be valid for computer generated displays. The latter assumption will not be valid if the graphics presents three dimensional views. In such cases, the distortion will depend on the amount of deflection used, with smaller deflections giving less distortion.

Effects of lag compensation with head position prediction and image deflection

The image deflection alone overcame the performance degradation due to time delays, but the simple head position prediction algorithm significantly reduced the amount of deflection required (Figures 10 and 11). This reduction in required image deflection reduces the restriction on the field-of-view and the image distortion imposed by deflection. Because the target was always captured near the centre of the field-of-view, as explained above, the benefits of this reduction were invisible to the subject and are not apparent in the tracking performance measures obtained from the experiment.

Conclusions

Head tracking performance was significantly degraded by lags greater than, or equal to, 40ms (excluding 40ms system lag). The finding that such short lags degrade head tracking performance has important implications for the future development of head-coupled systems.

Image deflection significantly improved tracking performance in the presence of time delays up to 380ms and performance was restored to that without a time delay.

The amount of image deflection required to compensate for the lags increased with increasing lag and there was a corresponding reduction in the field-of-view. For lags less than 280ms, the use of a simple head position prediction algorithm significantly reduced the amount of image deflection required and would therefore reduce any image distortion and restriction on the field-of-view.

Although the use of head position prediction to enhance the benefits of lag compensation by image deflection has been demonstrated, further studies to optimize the prediction algorithm are required.

Acknowledgements

References

1. Birt, J.A. and Furness, T.A. (1974) Visually-coupled systems. Air University Review, 20, 28-40.

2. Thompson, S.L. (1987) Virtual world of the future cockpit. Air and Space, April/May, 74-83.

3. Allen, J.H. and Hebb, R.C. (1983) Helmet-mounted display feasibility model. Advanced Simulation Concepts Laboratory, Naval Training Equipment Center, Report TR-NAVTRAEQUIPCEN IH-338, February 1983.

4. Woodruff, R.R., Hubbard, D.C. and Shaw, A. (1986) Comparison of helmet-mounted visual displays for flight simulation. Operations Training Division, AFHRL, Williams Air Force Base, AZ 85240-6457, U.S.A. Display, October 1986, p.179.

5. Wells, M.J. and Griffin, M.J. (1984) Benefits of helmet-mounted display image stabilisation under whole-body vibration. Aviation, Space and Environmental Medicine, 55, 13-18.

6. List, U.H. (1983) Nonlinear prediction of head movements for helmet-mounted display. AFHRL Technical Paper 83-45. Operations Training Division, Williams Air Force Base, Arizona.

7. Albrecht, R.E. (1989) An adaptive digital filter to predict pilot head look direction for helmet-mounted displays. Unpublished M.Sc. Thesis, University of Dayton.

8. Wells, M.J. and Griffin, M.J. (1987) Performance with helmet-mounted sights. Technical Report 152, Institute of Sound and Vibration Research, University of Southampton.

9. Williams, J.N. (1987) A pilot simulation investigating handling qualities and performance requirements of a single-pilot helicopter in air combat employing a helmet-driven turreted gun. M.Sc. Thesis, Naval Postgraduate School, Monterey, California, Sept. 1987 (AD-A186878).

Appendix

Image deflection: benefits

To illustrate the principle of the image deflection technique, consider an application in which a head-coupled system is used to present images captured from a head-slaved camera which follows the line-of-sight of the observer with a time delay. Figure A1 shows the effect on the displayed image when the head of the observer moves to acquire a stationary target at a constant angular velocity, $\dot{\theta}_h$ assuming the head-slaved camera follows with a constant time delay, τ.

After a time t, the head has traversed an angle of:

$$\theta_h = \dot{\theta}_h t \tag{1}$$

but the camera and hence images captured has only, moved by θ_c, where:

$$\begin{aligned}\theta_c &= \dot{\theta}_h(t - \tau) \\ &= \theta_h - \dot{\theta}_h \tau\end{aligned} \tag{2}$$

Therefore by deflecting the screen with an offset of $\dot{\theta}_h\tau$, the image is restored to its correct position .

Image deflection: restriction in field-of-view

Suppose the target is separated from the initial head position by an angle, θ_{ta}, and that the field-of-view (FOV) of the camera subtends an angle of φ. The image of the target will only be captured if it falls within the FOV of the camera:

i.e. $$\theta_{ta} - \theta_c \leq \varphi/2 \tag{3}$$

substituting θ_c from (2):

$$\theta_{ta} - \theta_h + \dot{\theta}_h\tau \leq \varphi/2 \tag{4}$$

If we rearrange equation (4), we get:

$$\theta_{ta} - \theta_h \leq (\varphi - 2\dot{\theta}_h\tau)/2 \tag{5}$$

where $(\varphi - 2\dot{\theta}_h\tau)$ is defined as the effective FOV, which is two times the maximum angular separation allowable between the target and the observer's line-of-sight so that the target will fall within display.

From equation (5), one can see that this effective FOV is reduced by any time delay present (τ).
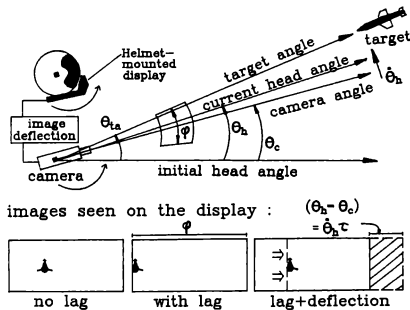
Figure A1   Diagramatic illustration of effects of time delays between head movement and image movement on images captured by a head-slaved camera. The effect of image deflection is also shown.

Unusual Attitude Recoveries Using a Pathway in the Sky

John Reising
Kristen Barthelemy

Cockpit Integration Directorate
Wright-Patterson AFB, Ohio

David Hartsock

Midwest Systems Research
Dayton, Ohio

ABSTRACT

Three different Head Up Display (HUD) formats were tested to see which would provide the pilot with the most effective means of recovering from unusual attitudes. Two of the formats were variations of conventional HUD formats, while the third utilized a Pathway-in-the-sky to guide the pilot back to the horizon. The conclusion was that, with adequate training, the Path performed as well as the more conventional HUDs, and provided the pilot with situational awareness by showing him the way to recover.

INTRODUCTION

Since their inception, Head-Up Displays (HUDs) have evolved from relatively crude gun and bombing sights to complex display devices containing information from a variety of sources. As the HUD progressed through various stages of development, the primary emphasis was on tactical operations; however, symbology was added to incorporate some of the elements of basic flight control and performance instrumentation. The result is HUDs which contain display formats that are almost, but not completely, adequate for flight in instrument flight conditions or for recovering from unusual attitudes. Of the two conditions just mentioned, unusual attitudes pose the more challenging problem and are the focus of the research presented in this paper.

An unusual attitude is an undesired aircraft attitude occurring inadvertently. It could result from a single cause or a combination of effects, such as, turbulence, pilot distraction, instrument failure, inattention, disorientation, or weather conditions. It is important to note that an unusual attitude is not defined by a particular value of flight path angle or pitch; for fighters in a dog fight an unusual attitude is much different from a transport aircraft flying the cruise portion of its mission, in other words it depends on aircraft type as well as flight phase. The inadequacy of defining an unusual attitude by a particular value of pitch/flight path angle is especially

true for agile aircraft such as the X-29 and X-31 which can sustain controlled flight at very high angles of attack (AOA)(1). Unlike conventional aircraft which do not sustain controlled flight above 30 degrees AOA and, consequently, can easily experience unusual attitudes in this region, highly agile aircraft sustain controlled flight in the post 60 degree angle of attack regime. Therefore, unusual attitudes for conventional aircraft can be different from those for highly agile aircraft.

In most instances unusual attitudes are mild enough to recover from by reestablishing straight and level flight using external visual cues and resuming a normal instrument cross check. Originally an unusual attitude was recognized in one of two ways: an unusual "picture" on the attitude director indicator (ADI) or unusual behavior of the performance instruments. With the advent of the modern fighter aircraft, HUDs have increasingly been used as the primary flight instrument; consequently, they too have become involved with recovery from unusual attitudes.

One problem with using the HUD in the area of unusual attitude recoveries is the symbology's inability to 1) convey to the pilot, in a timely and straight forward manner, that an unusual attitude really exists and 2) direct the pilot's recovery (2). In trying to determine one's attitude from the HUD, it is not always immediately apparent whether one is upright or inverted, climbing or diving, or if so, to what general extent because the pitch/flight path scales look about the same regardless of attitude. Generally on the HUD there is no clear distinction between sky and surface, the only difference being the type of lines on the pitch scale. The lines are solid for positive references to pitch or flight path symbologies and dashed for negative pitch and flight path symbologies. The overall pattern of the scales is symmetric about the horizon line which is not much longer than any other scale line. This lack of a clearly distinguishable horizon line further contributes to the problem of disorientation (3).

There have been a number of research efforts, both in the U.S. and the U.K., to redesign HUD symbology to create an asymmetrical picture on the HUD to give the pilot a better indication of positive

and negative attitudes (4). One of the major aims of these efforts is to give the pilot both a clear indication if he is in a nose high or nose low attitude, and the direction to the horizon when it is not in view. In achieving this goal various modifications to current HUD symbology have been attempted, resulting in pitch/flight path lines which are angled, tapered, or contain various horizon pointing indicators (5,6,7). Although modified HUD symbologies were used in this study, a completely different kind of symbology format, a Pathway in the sky, was also examined (8). The Pathway has been tested in simulations using it to depict complex curved paths the pilot may be required to fly in the terrain following/terrain avoidance mission (9). The results showed that when using the Pathway the pilot could fly with significantly better precision than when using conventional HUD symbology. The pilots also felt that the Pathway was a much more intuitive display since it could show the dynamics of the future path, something not possible with current HUD symbology. The current study extends the use of the Pathway to another kind of complex maneuver -- unusual attitude recovery.

OBJECTIVE

The primary purpose of this study was to determine which of three HUD symbologies was most effective in terms of interpretability and successful recovery from an unusual attitude. A secondary purpose was to determine if a nose high or nose low starting attitude affected performance, since the HUD symbology would present a different picture in each case.

METHOD

Subjects

Eighteen pilots from the United States Air Force, US Air Force Reserves, and other organizations at Wright Patterson Air Force Base participated in this study. One half of these subjects had previous HUD experience.

Design

The study utilized a 3x2 full factorial within-subjects design with two independent variables: (1) HUD type - Asymmetric, Global, or Pathway; and (2) starting attitude of the aircraft - nose high, or nose low. Each subject received a random order of eight unusual attitudes (four nose high and four nose low) for each of the three HUD symbology types. The order of presentation of HUD type was counterbalanced. Each subject,

therefore, had data collected for 24 trials.

Apparatus

Dynamic Cockpit. This evaluation was conducted in a generic fighter cockpit, roughly comparable in size to an F-15 fighter cockpit. A single multi-color CRT was used as the HUD, and its symbology was generated by an Silicon Graphics IRIS 3130. A side mounted force stick provided the pilot with control of the F-16 aeromodel flight characteristics. A speed brake control switch was provided on the throttle (See Figure 1).

HUD Formats. Three experimental multicolor HUD formats were tested. On two of the formats, the design of the flight path/pitch scales were variations of those found on a standard HUD, while the third used a command Pathway and "follow me" aircraft to lead the pilot through the preferred recovery procedure using the recovery rules described in Air Force Manual 51-37, Instrument Flying (10).

Asymmetric HUD Format. Each scale line was made up of two segments separated by a space through which the flight path marker traveled. The lines representing positive flight path/pitch angle were solid and color coded blue, while negative flight path/pitch angle lines were dashed and color coded brown (Figures 2 and 3). Each scale line had two horizon pointers on the inboard edge of each line segment. The length of each scale line was approximately one half the length of the horizon line. The positive scale lines always remained parallel with each other and with the horizon line. The negative scale lines formed an angle at the intersection of the horizon pointers and scale line elements. The resulting angle was equal to one half of the aircraft pitch angle. As the aircraft departed further from the horizon, the ever increasing angles formed a "directional tunnel" to the horizon. The scale lines were spaced and numbered at 5 degree increments from 0 thru 30 degrees and at 10 degree increments from 30 to 90 degree of flight path/pitch angle.

Global HUD Format. The Global HUD format had all features of the Asymmetric format except the pitch scale lines were longest (approximately 3 inches) near the real horizon and became progressively shorter (approximately 1/4 inch each) as pitch values increased either nose up or nose down. The 90 degree line was approximately 3/4 inch long. Thus, the pitch ladder was analogous to the latitude lines of a globe, i.e., the longest lines were at the horizon (equator) and the shortest lines at the zenith and nadir (north and south poles) (Figures 4 and 5).

Common Features of Asymmetric and Global HUDs. The positive (nose up) and negative (nose down) flight path/pitch scale lines used on the Asymmetric and the Global HUD format were separated by a horizon line. At aircraft flight path/pitch angles of up to 20 degrees (positive or negative) from level, the real horizon line was shown as a solid bold white line made up of two segments extending across the display from the airspeed scale to the altitude scale. At pitch angles greater than 20 degrees (positive or negative) a bold dashed white line (caged horizon) was shown at the top (bottom) of the flight path/pitch scale representing the direction of the real horizon. Numbers representing negative pitch were preceded with a minus sign. The scale lines were labeled on the right hand side only; when in an inverted attitude, they were on the left. A miniature aircraft symbol was displayed whenever the pitch attitude was greater than 30 degrees to provide pitch guidance and to counter the rapid movements of the flight path marker during dynamic maneuvering. The format background was black. The two formats also included a bank scale and pointer.

Pathway HUD Format. This format incorporated three main display elements: a unique horizon line, a Pathway, and a "follow-me" aircraft. (Figures 6 and 7)

The true horizon line was a long, bold, white line made up of two segments. As in the previous formats, the line showing the true horizon was solid, while the caged horizon, shown when the true horizon was out of the HUD's field of view, was dashed. At the ends of the horizon line were perpendicular bold lines which grew proportionally in length as pitch attitude departed from level flight. These "goalposts" were color coded: cyan for nose high and brown for nose low. A digital readout of the pitch angle value was placed above or below the right-hand portion of the goalpost.

In a previous study examining the goalpost as an aid to unusual attitude recovery, the subject pilots were able to recover as effectively with only goalpost symbology as with the complete pitch/flight ladder. However, the pilots believed that additional guidance symbology was necessary to build up situational awareness; therefore, it was decided to add the more intuitive Pathway symbology.

The Pathway consisted of a series of rectangular blocks combined in a specific sequence to form a route (highway). The blocks were drawn in perspective. Nominal dimensions were 300 feet in width, 500 feet in depth and one pixel thick with 1000 feet between blocks. Thirty six blocks were in view at all times. The blocks,

outlined in white, were transparent with a center line drawn on each block. The centerline was gray when viewed from above and medium gold when viewed from below the Pathway. The format background was black. The Pathway format displayed a follow-me aircraft, colored gray, which flew along the path. The position of this aircraft was beside the third block, left of the path with the aircraft's right wingtip on the left border of the path. The Pathway moved at a rate proportional to the speed of the pilot's aircraft.

Command Path Dynamics. The algorithms used to provide the path dynamics were based upon procedures described in Air Force Manual 51-37 for unusual attitude recoveries using attitude indicators: (1) "If diving, adjust power or drag devices as appropriate while rolling to a wings level, upright attitude and correct to level flight on the attitude indicator, (2) if climbing, use power as required and bank as necessary to assist pitch control and to avoid negative G forces. As the fuselage dot of the miniature aircraft approaches the horizon bar, adjust pitch, bank, and power to complete recovery and establish the desired aircraft attitude" (10, p. 2 - 16). These algorithms were used to control movements of the follow-me aircraft (pitch and roll) and the direction of the Pathway. These movements commanded a flight path to upright, straight and level flight. The subject's task was to fly formation with the follow-me aircraft while using the Pathway to anticipate turns and altitude changes. The flight path marker always indicated where the pilot's aircraft was heading. The horizon line and goalposts provided raw attitude data.

Common Features of All Three HUDs. Other elements of the formats were the same on each HUD. They included airspeed, altitude, heading, vertical velocity, acceleration (G's), Mach number, a flight path marker and a speed brake indicator. All formats were color coded: cyan referred to sky symbology, brown represented ground, information scales and horizon line were white while alphanumerics were shown in green. (Monochrome versions of all formats have also been developed.)

Procedure

Subject Briefing. Each subject was given the same briefing containing information on the purpose of the study, how to operate the controls in the cockpit, information about the various HUD symbologies, experimental procedures, and flight control training.

Training and Data Missions. The approximate time each subject participated in the experiment was one hour. After the briefing, the sequence of events for

testing each HUD symbology always remained the same. The pilot was given 10 minutes of free flight during which time he was to become familiar with the flight characteristics of the simulator, the symbology, and the response of the symbology with respect to control inputs. Following this, the pilot was given at least four practice unusual attitudes to recover from. Two started in a nose high attitude and two in a nose low attitude. This practice also allowed the pilot to become familiar with the testing procedure. Finally the pilot was tested on the symbology by recovering from eight unusual attitudes; four nose high and four nose low.

For practice and testing the CRT screen would start out black. A message would appear instructing the pilot to set the throttle to the midway point and release pressure on the stick. The purpose of the message was to ensure that all trials started at the same throttle setting and that an accurate measure was taken for the pilot's first input. A beep sounded, there was a one second delay, then the timer began as the display appeared with the aircraft in an unusual attitude.

The pilot's task was slightly different for the three types of symbology. For the Asymmetric and the Global HUDs, the pilot was to interpret the symbology and recovery to straight and level flight as quickly as possible. For the Pathway HUD, when the display appeared with the aircraft in an unusual attitude, path blocks automatically appeared in front of the pilot directing him to straight and level flight. Therefore, the pilot's task for this symbology was to follow the blocks to straight and level flight.

Each subject had 30 seconds to recover to straight and level flight. A successful recovery consisted of placing the aircraft within ± 2 degrees of level flight and ± 4 degrees of roll for two seconds, at which time the screen automatically went blank. After the pilot had completed all unusual attitude recoveries with a particular HUD format, he filled out a debriefing questionnaire.

Dependent Measures. Three dependent measures were collected: total recovery time, initial input time, and altitude change. Total recovery time was the time from display presentation until recovery to straight and level flight. Initial input time was the time from display presentation until the pilot made his first stick or throttle input. Altitude change was the difference between the initial altitude and the minimum or maximum altitude attained during the recovery from the unusual attitude. For

instance, for a nose low condition, the interest would be in altitude lost which is the difference between the initial altitude and the minimum altitude.

RESULTS

The data was analyzed using the Multivariate Analysis of Variance (MANOVA) Statistical Package for the Social Sciences (SPSS) program (Nie, Hull, Jenkins, Steinbrenner, and Bent, 1975). The results showed a main effect for HUD type ($F(6,66) = 9.15$, $p < .001$) and a main effect for starting attitude of the aircraft ($F(3,15) = 91.88$, $p < .001$). There was also a significant interaction between HUD type and starting attitude of the aircraft ($F(6,66) = 3.00$, $p < .012$).

DISCUSSION

The interaction effect between Nose High/Nose Low Attitude and HUD type can be explained in terms of the aeromodel used in this study. The F-16 "floats" at nose high attitudes and relatively low airspeeds, and these conditions were present in some of the unusual attitudes examined in this study. This effect is clearly shown when comparing the three dependent variables (Figures 8, 9, 10).

Initial input time and altitude change are virtually identical for Nose High/Nose Low Attitude within each HUD type, which is to be expected since these are not affected by the aeromodel. The third variable, total recovery time, is quite sensitive to the floating of the nose high attitudes, and this result is clearly reflected in figure 10.

A result more directly related to the effect of HUD type is shown in Figures 11, 12, & 13. The recovery times and the altitude change for both the Asymmetric and Global HUD formats were significantly different from the Pathway, and this trend was also shown in the initial input times. Why did this occur when the Pathway was supposed to be more intuitive?

A possible explanation for this difference could be familiarity; even though the pilots practiced with the Pathway until they felt they were familiar with it and could operate it efficiently, they had much more experience with the design and arrangement of the symbology of the other two formats since these formats were variations of HUD/ADI formats they had seen throughout their flying career, while the Pathway was completely new.

To investigate the familiarity issue further, a follow up study was conducted in which five of the pilots who had participated in the main study flew only the Pathway for five days. The same three dependent variables were also collected in this follow up study. The results showed substantial improvement in the pilots' performance. For example, at the end of five days, the pilots' total recovery time using the Pathway was 14.8 seconds (Fig. 14) which is 4 seconds faster than their

performance during the main study, and 2.6 seconds faster than the best of the other two formats. Similarly the initial input time after five days was 0.66 sec (Fig. 15); 0.2 sec. faster than the main study, and 0.09 sec faster than the best of the other two HUDs. Finally,the altitude change after five days was 4.2 thousand feet (Fig. 16); 1.4 thousand feet less than the main study, and 2 hundred feet less than the best of the other two HUDs.

CONCLUSION

The Asymmetric and Global HUD types might show further improvement with training, but the effect would be expected to be greatly lessened because of the pilots' extensive experience with these format types. The conclusion is that the command Pathway can offer a significant help to the pilot by showing him the way to recover from an unusual attitude, thereby preventing the confusion which often occurs in this situation.

REFERENCES

1. Borowski, R. & Reising, J. Cockpit Design Considerations for Highly Agile Aircraft, In Proceedings of the Sixth Symposium On Aviation Psychology, Columbus, Ohio, April, 1991.

2. McNaughton, G.B. (1985). The Role of Vision in Spatial Disorientation and Loss of Aircraft Attitude Awareness by Design. In Proceedings of the Aircraft Attitude Awareness Workshop (pp. 1-3-1 - 1-3-53). Flight Dynamics Laboratory, Wright-Patterson Air Force Base, Ohio.

3. Taylor, R. (1985). Aircraft Attitude Awareness from Visual Displays. In Proceedings of the Aircraft Attitude Awareness Workshop (pp. 2-7-1 - 2-7-17). Flight Dynamics Laboratory, Wright-Patterson Air Force Base, Ohio.

4. Zenyuh J., Reising, J., McClain J., Barbato D., and Hartsock D. (1987) Advanced Head-Up Display (HUD) Symbology: Aiding Unusual Attitude Recovery. In Proceedings of the 31st Annual Meeting of the Factors Society, New York, New York, Oct.

5. Reising, J., Zenyuh J. and Barthelemy K. (1988) Head Up Display Symbology for Unusual Attitude Recovery. In Proceedings of the National Aerospace and Electronics Conference, Dayton, Ohio, May,.

6. Deaton, J.E. and Barnes, M. (1990) An Evaluation of the Augie Arrow HUD Symbology as an Aid to Recovery from Unusual Attitudes. In Proceedings of the 31st Annual Meeting of the Factors Society, Orlando, Florida, Oct.

7. Taylor, R.M. (1984) Some Effects of Display Format Variables on the Perception of Aircraft Spatial Orientation. In Proceedings of the NATO AGARD Aerospace Medical Panel Symposium on Human Factors Considerations in High Performance Aircraft (pp. 14-1 - 14-14). Loughton, Essex, UK: Specialized Printing Services Limited.

8. Hoover, G.W., Cronauer,C.T., and Shelly, S.H. (1985) Command Flight Path Display F-14 Flight Test Program. (Tech. Report NADC-85128-60). Warminster, PA: Naval Air Development Center.

9. Reising, J., Barthelemy, K. and Hartsock, D.(1989). Pathway-In-The-Sky Evaluation, In Proceedings of the Fifth Symposium On Aviation Psychology, Columbus, Ohio, April, 1989.

10. Air Force Manual 51-37 (1984). Instrument Flying, Department of the Air Force, Headquarters US Air Force, Washington D.C.
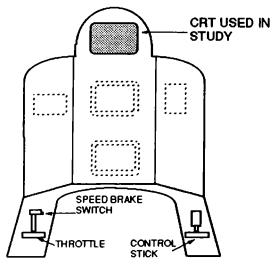
Figure 1. Cockpit Layout.
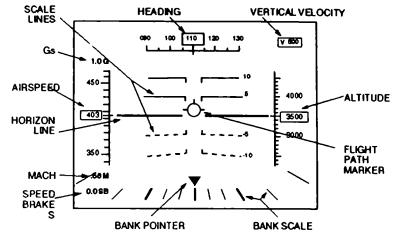
Figure 4. Global HUD showing a straight and level attitude.
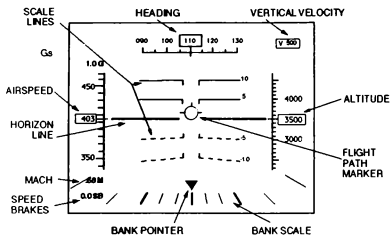
Figure 2. Asymmetric HUD showing a straight and level attitude.
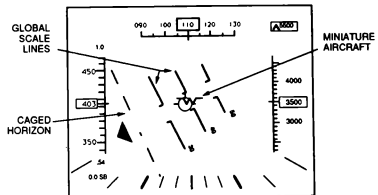
Figure 5. Global HUD showing an unusual attitude.
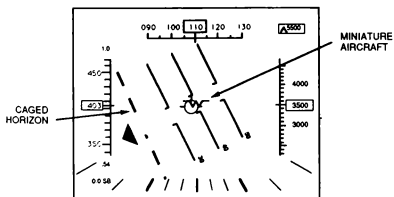
Figure 3. Asymmetric HUD showing an unusual attitude.
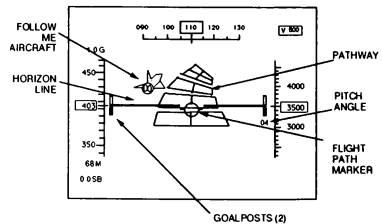
Figure 6. Pathway HUD showing a straight and level attitude with a predicted climb and right turn.
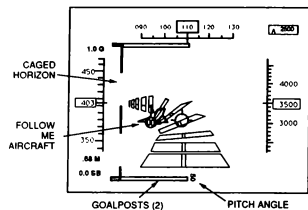
Figure 7. Pathway HUD showing an unusual attitude.

Figure 10. Altitude change as a function of HUD type and unusual attitude type.

Figure 8. Total recovery time as a function of HUD type and unusual attitude type.
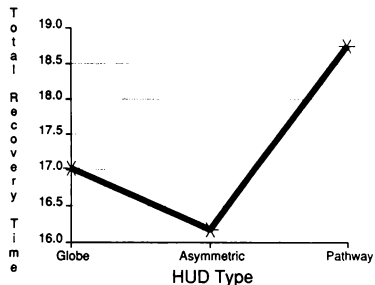
Figure 11. Total recovery time as a function of HUD type.

Figure 9. Initial input time as a function of HUD type and unusual attitude type.
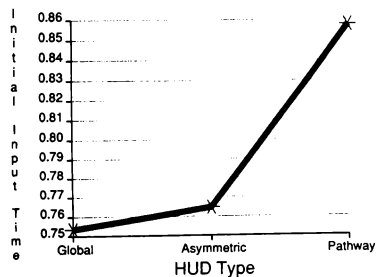
Figure 12. Initial input time as a function of HUD type.
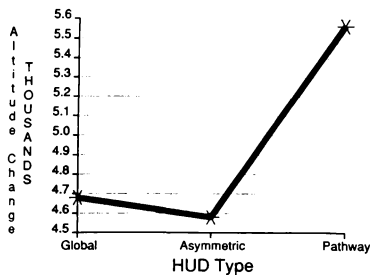
Figure 13. Altitude change as a
function of HUD type.
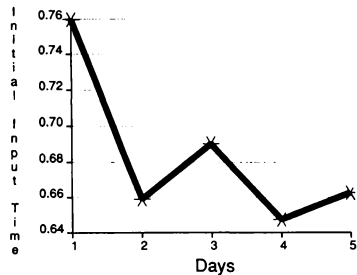
Figure 15. Pathway initial input time
as a function of practice.
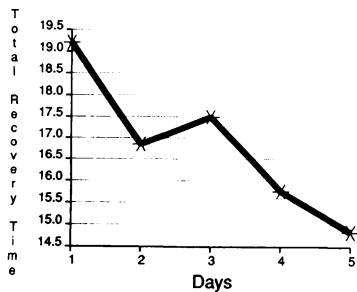
Figure 14. Pathway total recovery time
as a function of practice.
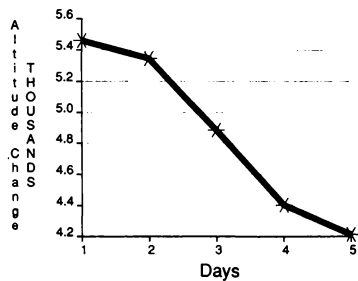
Figure 16. Pathway altitude change as
a function of practice.

TERRAIN MODELING FOR HIGH PERFROMANCE VISUAL SIMULATION

John Ellis and Ross Vellinga
Evans & Sutherland Computer Corporation
Simulation Division
Salt Lake City, Utah

ABSTRACT

This paper describes the sources of terrain modeling and rendering errors which occur with polygon-based visual systems. A new terrain modeling and rendering approach, developed at Evans & Sutherland and implemented in the ESIG®-4000 high-performance image generator, is described. Results of extensive testing over large geographic areas are presented. These tests demonstrate that this new approach, in which the image generator computes terrain polygons in real-time from a multiple level of detail terrain elevation grid, gives excellent terrain accuracy results, and offers significant improvements over previous methods.

INTRODUCTION

Many training tasks require computer generated imagery of terrain that is well correlated with other simulated sensors and with ground truth. Considerable attention has been recently given to improving the terrain accuracy of high performance image generators used in flight simulation.

This paper describes terrain modeling issues, including level of detail (LOD), and terrain model and terrain run-time errors, and includes the results of extensive testing of the ESIG-4000 approach to terrain generation.

TERRAIN LEVEL OF DETAIL

A multiple LOD terrain model is commonly used to reduce the overall terrain polygon load on the image generator by concentrating the polygons where they are most needed. This model is computed off-line by the database generation system. At run-time, as range from the viewpoint to terrain polygons increases, the image generator selects a coarser LOD version of the terrain model in that region. Numerous techniques have been developed, or proposed, to meet the conflicting requirements of minimizing both the displayed terrain positional errors, and the number of polygons used to display the terrain.

The optimal approach will involve rapid LOD roll-off, to reduce the number of polygons as a function of range, while at the same time retaining small polygons which represent fine topographic detail to long viewing ranges in places where such detail is needed. A terrain model with a large number of LODs will enable the LOD transitions to occur at closer ranges (thus conserving polygons), since the geometric differences between adjacent LODs will be small and thus less noticeable to the pilot.

As terrain LOD changes are made, the discrete nature of each LOD requires some type of blending to eliminate the sudden jumping or popping of terrain polygons at the LOD transition point. Previous systems have commonly used a "fade-level-of-detail" or FLOD approach. This entails computing and displaying both the finer and coarser LOD terrain over some transition range zone at each LOD change. The finer LOD terrain fades out and the coarser LOD terrain fades in over the transition zone.

FLOD has been used successfully for many years, but it has several

disadvantages. It is wasteful of IG polygon processing power, since two sets of polygons must be processed within each transition zone. In addition, this FLOD process is accomplished using a variable transparency blend, and transparency consumes additional CIG processing resources. Also, the elevation of a point on the terrain surface in each transition zone is indeterminate, since two different terrain geometries are being displayed simultaneously, leading to problems when features are placed on the terrain surface.

As an alternative to FLOD, algorithms for continuous LOD terrain have recently been developed. Continuous LOD terrain evolves through its LODs using a process from the animation industry known as "betweening." A finer terrain LOD region "grows" out of a surrounding coarser LOD region. First, new polygons are created by subdivision of the surrounding coarser polygons. Initially, each new polygon is created in the plane of the polygon which surrounds it. Then, as the polygons' vertices progress through the transition zone, the Z values of the polygons' vertices gradually move up or down to their next LOD value.

Continuous LOD has these advantages over FLOD:

• Only one set of polygons is used to represent the terrain skin, even in LOD transition zones. Thus fewer polygons are used and elevation ambiguity is eliminated.

• LOD transitions can occur as a function of both terrain roughness and range, therefore allowing large polygons (coarse LOD) to represent smooth terrain, even in the scene foreground, while enabling small polygons (fine LOD) to represent rough terrain at long ranges.

These advantages result in significant savings in terrain polygons while also providing superior terrain accuracy.

TERRAIN SOURCE DATA

An elevation grid, such as Digital Terrain Elevation Data (DTED) supplied by the Defense Mapping Agency (DMA) or the USGS is the most common form of terrain source data used in visual simulation.

In many military flight simulation applications, the correlation between the terrain as rendered by the visual system and the radar image generator is of prime importance. Most radar simulators use DMA DTED, and in practice DTED is usually considered to be "ground truth" when measuring correlation errors.

Elevations extracted from stereo aerial photographs or satellite scenes may also be used for terrain databases, and are usually provided in grid post form. This is often called a Digital Terrain Model or DTM.

An alternative representation is a Triangulated Irregular Network or TIN. This approach attempts to capture the terrain shape as an irregular network of triangles. However, terrain source data is not widely available in this format. Furthermore, the generation of a multiple LOD terrain model from a TIN is a complex, time-consuming process as compared to the trivial process required to produce a multiple LOD terrain model consisting of simple elevation grids.

TERRAIN ERRORS

Terrain modeling errors typically have two sources. The first is introduced between the source data and the finest LOD terrain model. This error can be measured by reconstructing the source data from the terrain model, and then measuring the elevation difference at each grid post between the source data and the corresponding elevation grid reconstructed from the terrain model. These will be referred to as off-line errors.

The factors which affect the off-line errors are:

• The spatial resolution of the terrain model grid

• The resampling method

• The terrain roughness

The source elevation grid can be reconstructed with less error from a 25 meter terrain model grid than from a 100 meter terrain model grid.

The resampling method used here was a simple bilinear interpolation using the four closest grid posts.

In areas of high roughness, the off-line errors tend to increase in magnitude, regardless of the terrain skinning method used.

The second source of errors is caused by polygon throughput limitations in the image generator, and will be referred to as *run-time* errors. As coarser terrain LODs are selected to reduce polygon processing load, fine detail is lost, resulting in the displacement of topographic detail. The magnitude of this error may be found by measuring the displacement between the theoretical displayed position of a point on the terrain model at its finest LOD and the actual position using an appropriate LOD for the range and polygon load as computed by the image generator.

Factors which affect the run-time errors are:

- Number of displayed polygons
- Field of view
- LOD roll-off profile
- Range
- Terrain roughness

As the number of polygons used to render the terrain model increases, finer LODs can be used to longer ranges resulting in a closer approximation to the true shape of the terrain, and thus lower run-time errors. If the polygon capacity of the IG were unlimited, then the finest LOD terrain could be used from the eyepoint out to the visibility limit, and the run-time terrain errors would be zero.

Decreasing the field of view, while rendering the scene with a constant number of polygons concentrates these polygons over a smaller area of the database resulting in a better terrain fit and thus lower run-time errors.

The LOD profile is an IG specific characteristic which determines where (at what ranges) topographic detail will be concentrated. The image generator used in this study was the Evans & Sutherland ESIG-4000. This image generator has up to 12 terrain levels of detail and numerous bias factors which can change the LOD profile, as required for different applications.

The transition range for a given terrain polygon is determined off-line as a function of terrain roughness and on-line as a function of polygon load.

LOD roll-off usually results in larger absolute errors as range to a terrain polygon increases. However, because the range is increasing, the pilot will perceive range errors as a relatively constant angular displacement between the displayed and ideal terrain at different LODs.

Terrain roughness affects accuracy because rough terrain requires more polygons to represent it to a given error than smooth terrain.

We have measured the terrain errors in a number of different ways, but have found perceived angular errors, expressed in milliradians, and range and roughness based errors, expressed in feet per nautical mile per unit of standard roughness, to be the most comprehensible. These error measurement criteria are explained in the next section.

EXPERIMENTAL RESULTS

Three contiguous areas of DMA DTED were used to measure the accuracy achievable with the ESIG-4000. They are illustrated in Figure 1. *The total area is 196 geo-units, which represents, in the opinion of the authors, a sufficiently large sample area from which to draw meaningful conclusions.*

Figure 1. Test Areas Used to Measure Terrain Accuracy
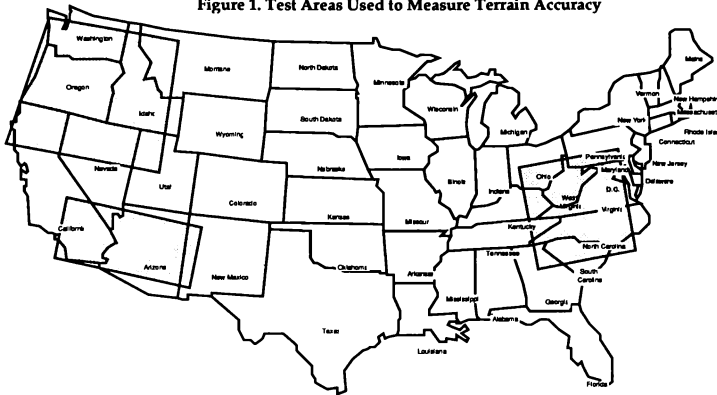
Standard Roughness

The U. S. Air Force has defined a measure of Standard Roughness for terrain. It is computed from DMA DTED in 61 x 61 grid post patches. For each elevation in the patch (excluding the perimeter values), the difference between the elevation and the average of the four adjacent elevations in latitude and longitude are computed. A latitude based convergence correction factor is applied. The absolute value of the differences is summed and divided by the number of processed values per patch (59 x 59).

Instantaneous roughness values may be extremely high, for example near a cliff, but the roughness value computed for each patch usually does not exceed 10, even in mountainous terrain. When the terrain is flat, or when it has a constant slope, the roughness is zero.

Roughness is included in terrain accuracy measurements because in rough terrain more polygons are needed to maintain a given terrain error.

Off-line Errors

The off-line errors were measured using the following procedure.

The elevation grid posts within each geo-cell were resampled using bilinear interpolation from the geodetic coordinates of the DTED into a Cartesian coordinate system suitable for the ESIG-4000. The mapping function produced an address in the DTED grid for each elevation post in the output grid. The bilinear interpolation computed the elevation of the output grid post from the four closest DTED elevation grid posts.

The spatial resolution of the resulting grids was computed at 25, 50 and 100 meters. Then these grids were resampled back to geodetic coordinates to produce an elevation value at the same location as each grid post in the DTED.

This technique could also be used to measure terrain accuracy of TIN based systems. A DTED equivalent elevation grid could be reconstructed from the finest LOD of the terrain model and the result compared with the source DTED.

The absolute value of the differences between the DTED elevation posts and the

reverse transformed ESIG-4000 elevation grid posts were summed over 61 x 61 grid post patches and divided by the standard roughness (as defined by the U.S. Air Force) of the patch and the number of samples. This results in an average error per unit of standard roughness, which gives a good sense of the overall terrain fit.

The results show that the average error in meters per unit of standard roughness is 0.551 for a 100 meter grid, 0.277 for a 50 meter grid and 0.120 for a 25 meter grid. This means, for example, if the terrain roughness is 3 and a 50 meter grid is used, then the average error will be 0.831 meters between the DTED source and an equivalent

DTED grid resampled from the finest LOD terrain model.

These errors are insignificant when compared to the DMA DTED accuracy objectives of ± 30 meters at 90% linear error absolute vertical and ± 50 meters at 90% circular error absolute horizontal.

The elevation differences were also used to compute 90%, 95% and 99% maximum errors. The 90% error is that value, in meters, which is less than or equal to the error measured in 90% of all the points tested. These numbers give a good sense of the absolute terrain fit. Figures 2, 3, and 4 show cumulative histograms of the errors.

Figure 2. Cumulative Histogram of 90% Off-line Errors
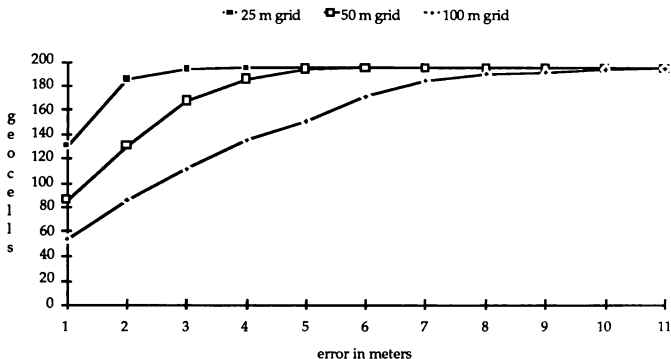
Figure 2 shows, for example, that in 185 of the 196 geo-cells tested, 90% of the maximum errors were less than 2 meters when a 25 meter grid was used for the finest LOD, while the same error was produced with 130 of the 196 geo-cells when a 50 meter grid was used, and 87 of the 196 geo-cells when a 100 meter grid was used.

The cumulative histograms also show that when a 50 meter grid was used for the terrain model finest LOD, 90% of the maximum errors in all geo-cells tested were less than 7 meters, 95% of the maximum errors were less than 9 meters and 99% of the maximum errors were less than 15 meters.

143

Figure 3. Cumulative Histogram of 95% Off-line Errors

·■· 25 m grid   ▫ 50 m grid   ··· 100 m grid
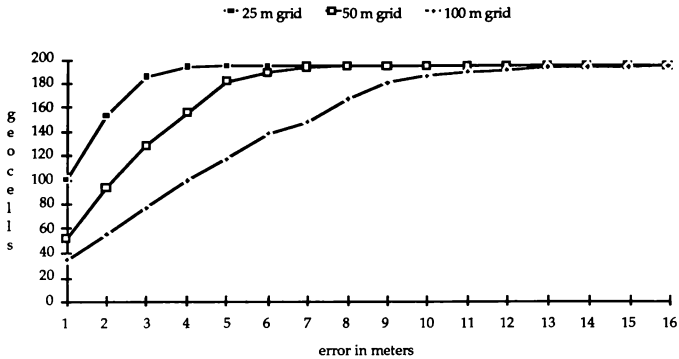
Figure 4. Cumulative Histogram of 99% Off-line Errors

·■· 25 m grid   ▫ 50 m grid   ··· 100 m grid
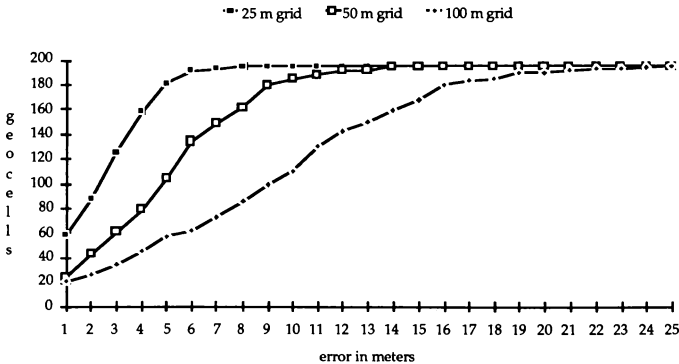
Run-time Errors

Next, we will examine the run-time errors *which are generally much larger than the off-line errors.* The run-time errors were measured using the ESIG-4000 emulator. Tests were run using several fields of view and terrain polygon counts. The mean number of displayed polygons for the 450 viewpoints tested was 1218. The FOV was 60 x 45 degrees, and the errors

were computed from the eyepoint out to a range of 80 kilometers.

For each test case, the emulator terrain processing module generated the vertices of each rendered terrain polygon. From this data, the equation of each terrain polygon's plane was found.

Then the intersection between each grid post, at the terrain finest level of detail, and the rendered terrain polygon was found. The magnitude of the resulting elevation errors, along with the range to each point, and the standard roughness of the corresponding DTED patch were calculated.

From these results several ways were developed to express the run-time errors. First was an angular error, measured in milliradians, between the displayed surfaces, and the surfaces which would have been displayed had the finest LOD terrain been carried out to the extent of the visibility range. These errors were collected for the 90% and 99% error levels from 450 viewpoints and are illustrated in Figure 5. This cumulative histogram shows that the error is less than 10 milliradians at the 90% error level for the vast majority of the eyepoints measured.

Figure 5. Cumulative Histogram of Run-time Angular Error

··■· 90% angular error   ◻ 99% angular error

error in miliradians
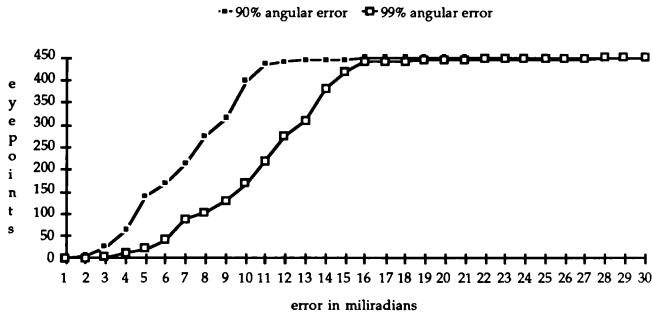
Another way to express run-time errors is the ratio of absolute error (in feet) to range (in nautical miles) divided by the standard roughness (as measured in 61 x 61 patches of DTED). These errors, for the same viewpoints, at 90% and 99% error levels, are illustrated in Figure 6. This cumulative histogram shows, for example, that the error is almost always less than 20 feet per nautical mile per unit roughness at 90% error level.

Figure 6. Cumulative Histogram of Run-time Error in ft/nmi/R

·■· 90% error ft/nmi/R   □ 99% error ft/nmi/R

error in ft/nmi/R
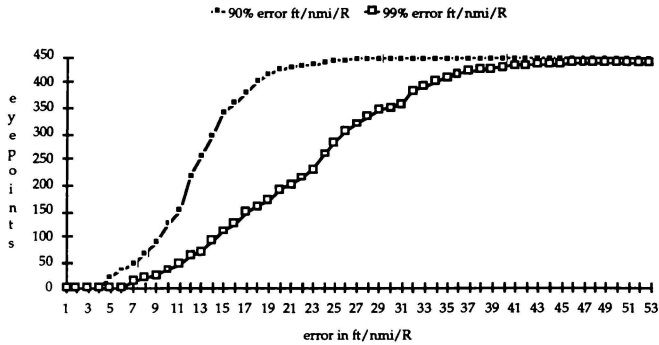
The effect of LOD on displayed terrain is illustrated in figures 7 through 10. Figure 7 shows a scene with 1400 terrain polygons, while figure 8 shows the same scene displayed using only the finest LOD, resulting in 25,000 terrain polygons.
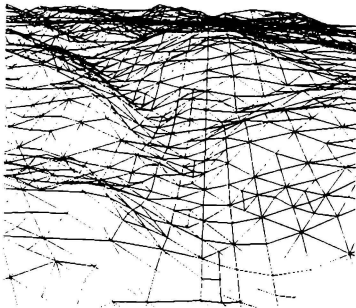
When photographic texture is applied to these scenes, the results, in figures 9 and 10, show how little meaningful detail is lost after the level of detail calculations reduce the number of displayed polygons by 95%.

Figure 7. 1,400 Polygons

Figure 8. 25,000 Polygons
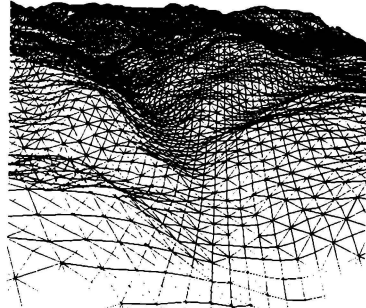
Figure 9. 1,400 Polygons   **Figure 10. 25,000 Polygons**
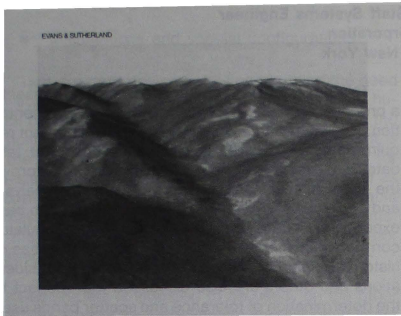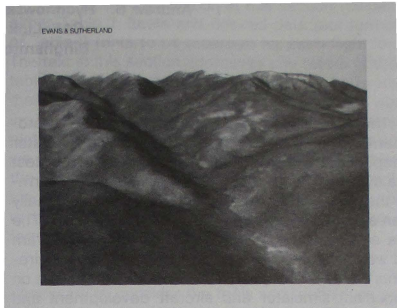
CONCLUSIONS

A large number of test cases were generated in order to produce enough data points to estimate the off-line and run-time errors which would occur in a production sized database.   These results were measured over three very large geographic areas and represent meaningful data rather than a single viewpoint within an isolated patch which happened to exhibit good accuracy.

Our results show that continuous terrain LOD produced from a regular grid, multiple-LOD terrain model gives the following advantages:

• Straightforward database processing enables rapid off-line generation of the terrain model from the elevation source data.

• Negligible errors between the source data and the finest LOD terrain model. For example, a 50 meter grid ESIG-4000 database will differ from the DTED source by an average of less than a meter, for terrain of roughness less than 4.

• Run-time errors typically less than 10 milliradians, for 90% of all errors measured, which is equivalent to an error of less than 20 feet/nautical mile/unit standard roughness.

FLIGHT TEST PROGRAM FOR ANALYSIS AND VALIDATION OF HELICOPTER SIMULATOR AERODYNAMICS

Susan C. Garing*, Senior Systems Engineer
Andrew S. Rychnowski, Staff Systems Engineer
CAE-Link Corporation
Binghamton, New York

Abstract

The time required to design and validate the aerodynamic code for a helicopter pilot training simulator is dependent on the quality and quantity of flight test data available to the program. Obtaining and confirming the consistency of this data set has historically been a problem for the simulator manufacturer. The data sets available predated the simulator program and were oriented toward the demonstration requirements for the helicopter itself. The new emphasis on concurrent simulator and aircraft development and greater simulator fidelity has created the need for dedicated simulator-oriented flight test hours. In this paper suggestions are made for flight test data processing and gathering intended to improve the utility of test data for simulator analysis and correlation. An outline is presented for a dedicated flight test program consistent with simulator development requirements. The program is designed to support both simulator aerodynamic model analysis and training simulator acceptance. A rationale for performing these tests is given. Parameters to be recorded, such as aircraft attitude, are enumerated for each test. The paper also describes alternate subsets of the specified parameters and tests which will serve as test data requirements when limitations in budget or time restrict the availability of the complete preferred data set.

Introduction

The new emphasis on concurrent simulator and aircraft development and greater simulator fidelity has created the need for dedicated simulator-oriented flight test hours.

The design and trainer acceptance effort is limited to time domain, handling qualities dynamics and statics. The models being developed are not intended for high-fidelity engineering research, accident investigation, or control law development. Aerodynamic model design complexity is limited by a requirement to run in real time (16–60 Hz) and to share processor space with a myriad of tactical, navigational, and training systems tasks which are necessary for total mission training.

_____

* AIAA Member

This paper outlines lessons learned and presents a proposal for flight test data collection and presentation consistent with training simulator development requirements. It describes techniques for flight test data processing and gathering intended to improve the utility of test data for training simulator analysis and correlation. These techniques are based on past experience regarding flight test data and simulator correlation problems and experience. Major areas of historical difficulty are addressed. These considerations and others are also discussed with respect to the determination of tolerance and scatter bands used when quantitatively validating training simulator performance.

The flight test program is designed to support both simulator aerodynamic model analysis and training simulator validation. The proposed plan reflects past experience with the realities of flight test correlation of the simulator, and the realities of flight testing.

Design Versus Acceptance Criteria

For military simulators, the Criteria Report contains flight test data against which the training simulator aerodynamics performance must be compared for validation purposes. In defining which tests and parameters are included in the validation set, consideration is given to the relative effectiveness of quantitative flight test data matching versus the more subjective pilot evaluation effort.

The validation data set is used to confirm the steady state and dynamic performance of the simulator throughout the flight envelope. In general the data runs cover:

- Weight and balance characteristics

- Flight control system mechanical characteristics

- Automatic flight control system characteristics

- Performance and trim data

- Stability and control characteristics

- Engine characteristics

- Ground handling

- Special flight regimes: hover, autorotation, weapons effects

- Malfunctions and special configurations

Test program design must keep in mind the need for automated test guide development – i.e., no human pilot required for performance of the test on the simulator. The critical nature of this requirement is in the need to repeatedly "validate" the simulator for on-site acceptances of individual devices as well as follow-on upgrades. The list of flight tests in Table 1 may not fully satisfy formal validation requirements, or the requirements for specialty areas of simulator research and development.

Design data is intended to be used, on the other hand, as a basis for aerodynamic and aircraft system model design. The scope of this data is much broader than that of the trainer acceptance criteria data, yet includes that data. It can include complex maneuvers that are used in-house to investigate and confirm transitions and trends between steady-state regimes, or to gain insight into rotor/fuselage/stabilator interference effects, etc. Although this design data is not used for acceptance evaluation, it is nonetheless valuable information in closing gaps in the aerodynamic model. It is assumed that the flight test data is supplemented by analytical and windtunnel data for initial model component design (e.g., fuselage maps). In addition, basic weight and balance model and configuration data is required.

Unfortunately, since most new simulator projects run concurrently with aircraft development, flight test data collected from Preliminary Aircraft Evaluations (usually available in the early stages of a project) may reflect a version of the aircraft which differs from the production run model. Because of the lead time required for simulation development, the simulator manufacturer usually has to impose a data freeze, which, for lack of any better data available at the time, includes this data.

All of these areas require close cooperation between the agency producing the flight test data, the airframe manufacturer, the simulator manufacturer, and the customer. This close cooperation and team effort during test data acquisition is important because accurate analysis and the consistency of flight test data is critical to the timely completion of a simulator to flight test data matching effort. The simulator manufacturer must be an active member of the team.

Data Set Requirements

Introduction

Most of the information in this section resides in either Table 1 (tests and desired data plot formats) or Table 2 (data to be recorded for each flight test). The text in this section addresses and refers to these tables. It is important for the reader to understand that Table 1 is what we (the simulator engineers working the flight test matching problem) would like to see plotted in future reports and Table 2 represents what we would like to see recorded and made available by the flight activity doing the testing. Again for emphasis, these requirements may not be adequate for other types of simulator data gathering or validation efforts outside the realm of training simulators.

When deciding which flight tests to perform, it is necessary to consider the tools and techniques used to develop and validate training simulation math models. At Link we employ "trimmers" and "correlators" for evaluating static data. These in turn rely on flight test data for full definition of static aircraft conditions. Successful model design in many regions of the flight envelope can depend on the availability of this data.[1] Typically data is missing in the center and edges of the flight envelope. These regions respectively represent both the very mundane and the very hazardous aircraft performance areas. The even spacing of aircraft and environmental test conditions, as recommended in Table 1, is intended to close the gaps in the middle of the envelope.

Specific tests of extreme conditions, including maximum velocity, center of gravity, and density altitude/outside air temperature, are listed to aid the simulator manufacturer in rounding out the envelope and demonstrating performance in these regions. Aeroelasticity, reverse flow, and tip mach number effects, etc., make modeling difficult at the envelope extremes; however, simulator training tends to gravitate to these areas with much greater frequency due to the safety under which these conditions can be trained in the simulator.

Recommended Parameters To Be Recorded for All Tests

Parameters traditionally provided in aircraft handling qualities reports are not sufficient for simulator design and validation. Tolerances for measurement of flight test data parameters must be well within tolerances specified in the simulator specification and validation reports. The description of each test in the next

TABLE 1 FLIGHT TESTS AND MINIMUM DESIRED DESIGN DATA

* = a), b), c), d), e), f), g), n), o), r), and ee) data from Table 2

Engines

Engine Dynamics: time histories for engine conditions that result in aircraft handling qualities effects.

Parameters: *,q,p,h,j,k,l,v,ee,dd,ff

Response to Engine Loss (multiple engine helicopter):1 GW, 1 Altitude.

Parameters: *,q,p,h,j,k,l,v,ee,dd,ff

Pitot Static System

Calibration of Airspeed/Altimeter: 1 GW, Mid CG, cover maximum rearward, sideward, forward velocities and sideslips, over full altitude range and climbs/descents.

Parameters: m,h,j,k,l,n,o

Ground Handling

Ground Taxi, Braking: 3 GW for mid CG, and 3 CG's for 1 GW, aircraft attitudes for various wind speeds. Power to start taxi. Power for steady taxi. Power, torque pedal position required to break from ground (skids).

Brake force versus ground speed acceleration (wheels).

Parameters: *,j,k,l,v,gg

Statics

Slow–Speed Flight

Sideward flight: 3 GW, 3 CG, OGE Sweep to maximum sideward velocities in 5kt increments.

Parameters: *,h,i,j,k,l,n,o,r,v,aa

Rearward/Forward flight: 3 GW, 3 CG, 3 altitudes. Maximum rearward to 40 knots forward.

Parameters: *,h,i,j,k,l,n,o,r,v,aa

Critical Azimuth: wind in 15 degree azimuth increments. "worst case" GW and CG for control margin.

Parameters: *,h,i,j,k,l,n,o,r,v,aa

In–Flight Performance

Hover: 3 IGE and 3 OGE tethered (power) and free (rotor efficiency), mid CG.

Parameters: *,h,i,j,k,v,aa

Vertical Climb: 0 – 1500 AGL, 3 GW, mid CG

Parameters: *,h,i,j,k,v,aa

Level Flight Power And Trims: 3 GW, 3 CG, 3 altitudes. Sweep 40 knots to Vmax

Parameters: *,h,j,k,l,v,w,aa,dd

Climb/Descent: 3 GW, 3 airspeeds, 3 altitudes, engine torque increments 5% up to +/- 30% from trim. Include best rate and best angle.

Parameters: *,h,j,k,l,v,w,aa,p

TABLE 1 FLIGHT TESTS AND MINIMUM DESIRED DESIGN DATA (Cont'd)

Control Surface Trim Sweeps: (stabilator) 3 GW, 3 Airspeeds.

Parameters: *,h,j,k,l,v,w,aa

Stability and Control

Static Longitudinal Stability: 3 GW, 3 CG, 3 trim airspeeds +/- 5, 10, 15 knots increments (include hover), 2 altitudes

Parameters: *,h,j,k,l,v,w,aa

Short-Term Dynamic Longitudinal Stability: 2 GW, 3 CG, 2 Airspeeds (hover, cruise), 1 altitude, AFCS on/off.

Parameters: *,h,j,k,l,v,aa

Long-Term Response To Control Doublet Or Gust: at aircraft natural frequency. AFCS off/on.

Parameters: *,h,j,k,l,v,aa

Controllability - Control Axis: 3 GW, 3 CG, 2 Altitudes, 3 Airspeeds (hover, cruise, fast cruise), AFCS off. Steps up to +/- 2 inches.

Parameters: *,h,j,k,l,v,aa

Maneuvering Stability: 3 GW, 3 CG, 2 Altitudes, 3 Airspeeds

Parameters: *,h,j,k,l,v,w,aa

Static Lateral-Directional Stability: 3 GW, 3 CG, 2 Altitudes, 3 Airspeeds, AFCS on/off. Sideslips to +/- 20 degrees.

Parameters: *,h,j,k,l,v,w,aa

Dynamic Lateral-Directional Stability: 3 GW, 3 CG, 3 Altitudes, 3 Airspeeds, AFCS on/off. Sideslips to +/- 20 degrees. Include release from +/- 20 degree sideslips.

Parameters: *,h,j,k,l,v,aa

Miscellaneous Characteristics

Autorotation Entry: 3 GW, 1 CG, 3 airspeeds, 1 altitude.

Parameters: *,h,j,k,l,p,v,aa,ee,ff

Autorotation Flare: 3 GW, Airspeeds (min, max), 1 CG, 1 altitude.

Parameters: *,h,i,j,k,l,p,v,aa,ee,ff

Autorotation'Descents: 3 GW, 3 rotor speeds, 3 airspeeds (min - max). Need rotor speed versus density altitude for full low collective at minimum, middle and maximum autorotation airspeeds. Also rotor speed versus rate of descent trim sweep, airspeed versus rate of descent trim sweep.

Parameters: *,h,j,k,l,p,v,w,aa,ee

Blade Stall Boundary: as a function of airspeed, density altitude, loading.

Power Settling Characteristics: 1 GW, 1 altitude as a function of airspeed, density altitude, loading.

Parameters: *,h,j,k,l,v,aa

Mission Maneuvers

Time histories of maneuvers particular to the aircraft mission.

section highlights the parameters that should be recorded for that test. Parameters denoted by "*" in Table 2 should be noted in the header of all test results. Data should be available on magnetic tape (dynamics, statics) or in hardcopy tables (statics).

Recommended default tape format is ASCII, 1600 bpi, no headers or labels, with a hardcopy index of files.

Discussion of Individual Tests

Pitot Static System

This data is needed for design of the simulated cockpit instrument readings rather than as an aid to analysis of flight test data. It is important when testing with pilot in the loop that cockpit indicated airspeed and other indicators closely mimic the characteristics of the real test aircraft. Therefore the pitot system time lags, and errors due to orientation and rates, should be documented for the flight test aircraft.

Ground Handling

Braking: Brake force versus ground speed acceleration (wheels).

Ground taxi: Control positions and aircraft attitudes for various wind speeds and ground speeds. Power to start taxi. Power for steady taxi. Power and torque pedal position required to break from ground (skids).

Ship taxi: angle of ship pitch and roll required to cause helicopter roll or slip, with full low collective.

Ground handling is often a region left to pilot opinion. This data would establish the preliminary strut, tire and brake model coefficients. These are not usually provided by the manufacturer.

Statics

It is important for static trims that all control system rigging relationships be known. How does the automatic stabilization system affect static flight cases? Are there active biasing devices in the control system, for example, that respond to aircraft attitude? Are the gains and offsets used by the flight test aircraft equivalent to the gains used by the production aircraft?

Wherever possible, when gathering model design data, the number of independent variables should be reduced. For example, freezing a moving stabilator at a fixed position with this position held constant across several types of tests.

Slow–Speed Flight – Aside from the standard parameters and tests listed in Tables 1 and 2, flight parameters or special tests which show any particular anomalies in this region should be identified by the flight test activity. As an example, in sideward or forward/aft slow speed flight, specific handling qualities anomalies may occur due to tail rotor inflow problems, rotor wash on stabilator effects, a pedal stop being reached, or control authority for a particular center of gravity becoming bothersome. Acceptance pilots request that these special problems be simulated when they are identified on the actual aircraft. It is important for the simulator manufacturer to be aware of these kinds of problems (particularly if they are safety of flight situations) early in the design process.

Critical Azimuth – Critical azimuth data provides check data for fuselage aerodynamic maps and tail rotor performance at slow speed. Regions of increased airframe vibration should be identified and mapped during the flight test activity. This data is usually collected at a single airspeed, 30 knots. An additional test at 15 knots would provide an intermediate design point between hover and 30 knots. This would also provide verification of slow speed data collected in this regime.

In–Flight Performance

Hover – Tethered hovering flight provides the modeler with a somewhat simplified design condition. For mapped rotor models the total thrust and torque versus collective points are correlated using these data sweeps. In blade element rotor models the blade static drag and lift coefficient curves can be checked for a range of angles of attack.

Otherwise identical IGE and OGE tests provide data for the modeling contributions needed to simulate the wake interaction with the ground plane.

Vertical Climb – Vertical climb data provides insight into additional collective and rotor inflow versus torque and thrust points as well as providing correlation data for the fuselage drag maps if Table 2 type data sets are available.

Level Flight Power and Trims – Shaft horsepower is included with control trims, instead of as a separate topic, because correlation of the rotor torque and thrust must be matched with corresponding stick trim positions. These tests are almost never plotted together. The simulation engineer must try to match the flight conditions of control trim tests with SHP data.

TABLE 2 DATA TO BE RECORDED FOR EACH FLIGHT TEST

(Note: Asterisked items appear in test headers for all tests)

* a) Rotorcraft tail number
* b) External stores loading (store type and station)
* c) Landing gear position (if applicable)
* d) Gross weight
* e) Weight empty
* f) Inertial properties (at least Ix, Iy, Iz, Ixz)
* g) Center of gravity position (FS, BL, WL)
  h) Aircraft attitude (beta, alpha)
  i) AGL altitude
  j) Aircraft linear rates, rate of climb/descent, linear accelerations (load factor) in all axes.
  k) Aircraft angular rates, acceleration in all axes.
  l) Calibrated airspeed, true airspeed, ground speed (where applicable)
  m) Airspeed position error
* n) Density/pressure altitude (or range of altitudes)
* o) Outside air temperature
  p) Engine torque, FTIT, fuel flow, N_1, N_2
  q) Throttle position
* r) AFCS/stability augmentation modes engaged
  s) Ambient turbulence rating during testing.
  t) Sensor locations in terms FS, BL, WL. Location of sensors in the flight control system (e.g., control force, cockpit control position, surface position, etc.).
  u) Accuracy of sensors. Preferably based on actual calibration of the sensors used.
  v) Control positions – static and/or dynamic traces all axes
  w) Ball position
  x) Rigging, freeplay, trim deadbands
  y) Control surface schedules and rigging
  z) Control positions at the rotor hubs (main and tail rotor)
  aa) Control surface positions (stabilator), measured at surface
  bb) Average rotor flapping angles
  cc) Rotor brake used/not used
  dd) Rotor powers, thrusts, torques (all rotors)
* ee) Rotor speeds
  ff) Video recording of cockpit instruments
  gg) Wind speed and direction (ground reference tests)
  hh) Angle of attack and sideslip of fuselage
  ii) Angle of attack and total head of all aerodynamic surfaces (at midspan, on both sides of the aircraft)

The best available match is not always from the same test run and it may be of a significantly different gross weight or density altitude. This usually leads to few meaningful test cases, even from a large flight test report. In addition, the flight testing should be conducted on a single aircraft with gross weights at the light end, medium and heavy end of the range to be correlated. The extremes are very important for the simulator correlation with test data effort. Although pilots will not frequently fly under extreme conditions in the real helicopter, simulator training tends to gravitate to these areas with much greater frequency. To this end, higher airspeeds for level flight, out to V_H, should be obtained and the data recorded and identified at V_H. Typically, airspeed data is cut off anywhere from 10–15 knots too soon and good V_H data is not recorded or plotted.

Climb/Descent – Climb/descent data provides additional points for establishing math model torque and thrust trends. This data also provides insight into the fuselage drag map design and rotor–to–fuselage interference models.

Control Surface Trim Sweeps – The existence of non–rotor control surfaces on helicopters is a testament to their capacity to affect dynamic and static aircraft attitude and response. Windtunnel or theoretical data for beta +/– 180 and alpha +/– 90 degrees is required for basic model design of these surfaces. However, recognizing the complexity of the interactional aerodynamics between components of the helicopter, control surface trim sweeps are required to verify the design of simplified component interaction models.

These surfaces have created difficulties beyond those of aerodynamic modeling for the simulator aerodynamic model design engineer. Flight tests conducted using artificially fixed control surface positions, or employing rigging different from "production" aircraft design characteristics, can frustrate tolerance matching efforts if the abnormal control surface positions are not stated clearly. To aid correlation of control surface effects and verification of aerodynamic models for these surfaces, a complete pilot-stick-to-surface rigging diagram should be made periodically for each flight test aircraft. These rigging diagrams should be provided with the data sets. Also, measurements of actual surface deflections should be recorded for all tests.

Stability and Control

All aircraft dynamic tests must include time histories of all controls and resulting aircraft motions. Status of all automatic stabilization equipment must be listed with each test. It is typically assumed that the aircraft stabilization equipment is modeled on the simulator correctly within the frequency ranges of interest. Therefore AFCS–off characteristics are verified first for each axis. AFCS–on data should then "fall in line". Both AFCS on and off conditions should be used for dynamic tests and static tests if the AFCS status affects the results of those tests.

Perhaps even more critical for dynamic model design and verification, than for static model design, is the reduction of the design problem independent variables such as movable stabilization surfaces. Ideally the response characteristics of each dynamic element of the helicopter should be independently verified. Isolated component data is rarely available.

Static Longitudinal Stability – These tests are important for handling qualities evaluation and frequently are difficult tests to correlate due to a lack of full data about the trim conditions. Ideally these tests should confirm the speed stability of the rotor and the design of any stabilizing surface models.

Short–Term Dynamic Longitudinal Stability – Include a short–term response to control doublet at the natural frequency of the helicopter to show maximum effects of frequency and damping characteristics.

Long–Term Response to Control Doublet – This is a check case for unaugmented aircraft conditions. It is required to answer the question: should the AFCS on or off cases be free from phugoid–type oscillations? What does the phugoid look like? How does the AFCS affect it?

Controllability – Each Control Axis – These time history tests identify the control power and rate damping coefficients. Ideally non–rotor aerodynamic controls should be frozen during a portion of these tests so that the rotor response can be further isolated. It has traditionally been a difficult task matching flight test report plots of maximum acceleration, time to maximum acceleration, and rate, versus control displacement. As for other tests, all data should be available for each test, but here, in addition to time histories, reduced results should be presented. This allows the model designer to use the pilot's actual on-axis and off-axis control inputs rather than an idealized ramp input during the simulator validation effort.

The final analysis of aircraft response to controllability inputs must be done in the simulator with all hardware and software time delays in the loop. Experience has shown that final pilot tailored controllability is not always within the tolerances specified for matching actual aircraft data due to the perceptual differences between the simulated and actual environment. Advances in simulator technology, such as reductions

in throughput lags and improvements in visual systems, should minimize this problem.

Maneuvering Stability – Test results should include level turns, rate of descent, and constant collective turns. Symmetrical pullups are preferred for isolating model problems because of the reduction in cross-axis coupling effects (if the desired load factors can be maintained long enough to get quality data).

Maneuvering stability traditionally provides a check of pitch stability versus load factor. Turning tests also provide a check of main rotor lateral force for rotor map models, and fuselage forces for all models, as well as steady state angular rate correlation terms in pitch and yaw. The absence of complete aircraft fuselage velocity information (sideslip, rate of descent) and angular rate data has made the analysis difficult in the past.

Static Lateral–Directional Stability – Both static and dynamic lateral–directional stability data present the designer with the opportunity to test vertical tail, tail rotor, fuselage, and other lateral/directional aerodynamic force producers. The success of static tests such as these forms a basis for the performance of the simulator in even more complex maneuvers such as autorotational entry and mission specialized maneuvers. Tests should be performed by using steps in pedal.

Tail rotor models often employ a much simplified linear equation set, or simplified set of rotor maps. Accurate sideslip/control position data provide necessary checks and design points for extremes of tail rotor performance and the effect of independent variables. The need for these more specialized tests would be lowered if full sideslip and control position data were available on all trims and dynamic tests. In addition, accuracy of control margin in these axes can only be determined if full control position data is available (with appropriate aircraft rigging diagrams, if direct measurements of angles at the swashplate are not available).

Dynamic Lateral–Directional Stability – Doublets in lateral stick and pedal provide time history response data which indicates frequency, phase and damping response about all axes and between axes.

Miscellaneous Characteristics

Autorotation Entry, Flare, Descents – Autorotation is both a prime candidate for safe training in simulators and a difficult aerodynamic regime in which to satisfy test tolerances. It is a regime that lends itself to pilot-in-the-loop tailoring for entries, touchdowns, and recoveries due to the complexity of the maneuvers, and the subjectivity of pilot technique. Repeatable data gathered during autorotative trims is important in order to insure that the design is "in the ball park". Full data sets are especially important for model "debugging and fine-tuning". An accurate picture of the actual aircraft state is a must.

The regions of low collective pitch are shaped by the autorotation characteristics. Knowledge of the aircraft state at full low collective, and other lines of constant collective, gives an effective design tool. In addition, although model parameters are typically non-dimensionalized by rotor speed, sweeps of rotor speed are still required for model design and adjustment. Tests of density altitude versus rotor speed for various gross weights should be conducted for more than just one airspeed. Again, full control position plots should accompany these tests.

Blade Stall Boundary – Since this is a region for the pilot to avoid in actual flight, the boundary of the region is of valid training interest in the simulator. Unfortunately, as for other flow field and nonlinear aerodynamic phenomena, the flight conditions at which blade stall occur, and the pilot sensations associated with it, cannot be predicted sufficiently accurately by a real-time rotor model. Data is required to design and test model modifications as a function of the major stall boundary independent variables.

Other regions to be avoided by the pilot, such as droop stop pounding, should be identified experimentally, or estimated, and fully mapped.

Power Settling Characteristics – Like many other hazardous conditions practiced routinely in the simulator, power settling recovery and vortex ring state characteristics due to complex flow field phenomena do not necessarily "fall out" of the real-time simulator rotor math model. This type of highly nonlinear regime must be added to the basic model and therefore data is required for initial design. Validation is usually performed qualitatively with the pilot in the loop.

Mission Maneuvers

With the increasing emphasis on "Mission Rehearsal" using isolated and networked simulators of all types, there is an increasing need to validate simulator performance with the mission in mind. These maneuvers do not fit within the traditional handling qualities definitions. However, they may have been specified in the helicopter RFP and therefore they become design points. Examples of mission maneuvers include: the pop-up-and-shoot, quick stop, wingovers, return to target maneuvers, water autorotations, water taxi, etc.

Past Problems/Specific Lessons Learned

The task of flight test data correlation for helicopter training simulators has been complicated by several factors. These factors form the basis for our suggestions for future data gathering programs and requirements outlined in Tables 1 and 2. Their inclusion here is intended as a lessons learned type of summary based on our flight test correlation effort experiences. It's our hope that some of these problems can be worked out so that future work in these areas continues in an efficient and profitable manner.

The data necessary to design and validate a training simulator aerodynamic model, as defined in the introduction is rarely, if ever, found in a single flight test report, or gathered on a single particular aircraft. Mixing and matching data for a simulator acceptance criteria report or for a simulator trainer test procedure report only tends to increase the chances of inconsistencies. Data scatter within tests and between them makes correlation of data difficult and costly. Given that a flight test program will be conducted, inclusion of the following considerations in the program would seem to possess little additional risk or cost when contrasted with the benefits to the training quality of the simulator:

1) The data presented on any particular static data plot is usually too limited to identify the quality of the static trim in all axes (zero body axis acceleration condition). The simulator can be "trimmed" to match non-zero body accelerations, and other uncertainty-inducing anomalies, if they are indicated on the flight test data or are available using supplementary recorded data. Ideally, flight tests should be repeated if the data indicates a failure to trim.

2) Tolerance and calibration data on instruments used to measure flight parameters is usually not published. Adherence to specification requirements, which may be more stringent than the calibration of the test equipment which measured the parameter in question, needs to be addressed.

3) Dynamic data is reduced and modified before presentation. For example, controllability "maximum rate" point values are plotted rather than the time history of the maneuver itself. Lacking such primary data, it is very difficult to confirm that the test is set up on the simulator correctly. This adds the question: is the simulator model incorrect or is the simulator test setup wrong?" In addition, it is difficult to mimic complex pilot control inputs on the sim-

ulator, especially simultaneous inputs in more than one control axis.

4) Scattering of test conditions (e.g., altitude, airspeed) in different reports introduces an element of uncertainty in checking for model consistencies in areas not covered in the original report or in the areas covered by follow-on flight test reports. This is true especially if the follow-on test was conducted primarily for evaluation of a configuration change with perhaps a slightly different baseline configuration. This is particularly a problem if the original report did not cover the entire flight envelope to be matched. This also adds to the temptation (often the necessity) to combine tests from different flight reports for the acceptance criteria. The effect is to add to the data scatter problem. Merging test reports that use several different flight test aircraft to supplement acceptance criteria tends to add problems involving inconsistencies in the data to the data matching effort. These tests reflect different rigging, engine age, AFCS capabilities, etc.

5) Lack of data over large portions of the aircraft operating envelope (gross weight, center of gravity) restricts usability of data for design and design confirmation. If the gross weight or density altitude flight test data is not well distributed throughout the operational envelope, or does not cover the extremes, then model confirmation in these areas tends to be subjective in nature.

6) Tolerances for military helicopter acceptance (as spelled out in simulator specifications) generally do not reflect the level of difficulty of obtaining the flight trim point or the amount of scatter shown on the data plot. Uncertainty bands for trims in high rates of descent, and at slow speeds (which are more difficult to capture than level flight cruise trims) can exceed the bounds of standard contract specifications. These specifications are established prior to the evaluation or even the collection of this data. Again, closer cooperative efforts between the flight test activity, simulator manufacturer and customer are needed so that these items do not become bogged down as contractual compliance problems during the latter stages of simulator test and acceptance.

7) Performance of specific simulator-oriented tests should be addressed early in the aircraft

program prior to the flight test activity taking place. Delayed simulator–oriented testing results in extrapolation of the simulator model into areas of the envelope without any data for model confirmation. This tends to promote time–consuming subjective evaluations and re–evaluations.

8) Full instrumentation of test aircraft (see tables) would allow for additional data to be available for model confirmations as well as for inclusion in the acceptance criteria. For example, adding an angle–of–attack vane and pitot head to the leading edge of stabilizing surfaces at midspan would provide the modeler with interactive flow data between the rotors, fuselage, and surfaces. This data, if uniformly available on all tests, would allow insight into the resolution of a complex aerodynamic modeling problem by allowing sources of error to be better localized.

9) The use of consistency algorithms and redundant data by the data provider to improve the consistency and accuracy of the entire package prior to publication might be tried. Or in lieu of this, additional budget and authority for the simulator contractor to attempt this effort with the assistance of the data provider is needed. This is particularly important when resolving specific areas, for example, blending the high end of slow–speed flight parameters into the low end of level flight parameters.

10) Magnetic tapes and clean hardcopies of actual data need to be provided. Known errors should be noted on the data set as well as tolerances of instrumentation and details of reduction algorithms. Avoid smoothing of scattered data and non–dimensionalization of data.

11) Joint team participation by personnel representing the simulator contractor and the government needs to happen prior to the collection and data reduction of tests intended for simulator flight test matching. The more participation allowed up front in a simulator development program, the fewer problems are likely to arise out of poor data selection. This not only helps establish a concrete baseline prior to the design effort, but establishes a forum for problem resolution early in the simulator production life. The worst and most costly problems occur when data (or other requirements, for that matter) are identified as being incorrect after all the development, design,

correlation, in–house testing, and reports have been accomplished and this entire process must be repeated in whole or in part.

12) Data gathering on a specific aircraft which is representative of the fleet needs to be accomplished. Rigging diagrams, etc., for that aircraft need to be made available with the data set. Tolerances in data matching requirements should reflect expected variations among individual aircraft if more than one aircraft is used for data gathering.

13) Limiting the number of flight conditions over the range of different types of tests (e.g., gross weights, airspeeds, altitudes) so that comparisons between tests and trends are more readily established would significantly reduce the complexity, cost, and schedule of the correlation effort.

14) Flight test data gathering personnel should be identified and made available for ongoing resolution of concerns during the simulator test data correlation effort.

15) The original source data should be retained in a form that can be accessed many years after the data was taken and corresponding test reports were written.

Summary

The recommended test program ideas described in this paper, and the suggested validation methods and considerations, represent preliminary ideas. They are a step in the iterative process between the simulator manufacturer, the customer, and the flight test facility and pilots. The simulator validation effort, via validation with test data, cannot totally replace the "pilot–in–the–loop" evaluation of the simulator. Nor can it be rigidly defined at the start of a simulation project before the flight test data is available and a full analysis of that data is completed. An ongoing cooperative effort is required to assure that both the flight test data and the simulator sufficiently and accurately reflect the actual fleet aircraft. Teamwork at the very beginning should result in reduced validation cost and schedule without sacrificing simulator fidelity.

References

1. Garing, Susan C. and Michael L. Brychey, "Automated Flight Test Data Correlator for a Helicopter Flight Training Simulator", Presented at the 1987 I/ITSC Conference, Washington, DC, November 1987.

2. Humphrey, T., "Definition and Validation of the Flying Qualities and Performance Test Criteria for the Modern Operational Flight Trainer", presented at the 10th I/ITSC Conference, November 1990. Orlando, Florida.

3. Naval Air Systems Command, "Flight Test Data Requirements for Simulator Validation, Fixed-Wing and Rotary-Wing Flight Simulators", Draft Copy, October 1988, Washington, DC.

THE USER/OPERATOR ROLE IN
DEVELOPMENT OF A FLIGHT TEST DATA PACKAGE
FOR THE VALIDATION AND VERIFICATION OF A
FLIGHT SIMULATOR FOR COMMERCIAL
PILOT TRAINING AND CHECKING

Robert D. Foster[*]
USAir Inc.

Abstract

The simulator user/operator is the only party of all the various groups involved in a simulator program who is present throughout all phases of the program from pre-procurement through pilot training and continuing qualification of the simulator. This paper investigates the role of the user operator in conducting flight tests for data to support all these phases.

1. Introduction

Commercial operators of airplanes and flight simulators generally operate under Federal Air Regulation (FAR) Part 91, "General Operating and Flight Rules"; Part 121, "Certification and Operations: Domestic, Flag, and Supplemental Air Carriers and Commercial Operators of Large Aircraft"; or Part 135, "Air Taxi Operators and Commercial Operators." These FAR parts allow the use of approved flight simulators in lieu of airplanes for training and checking of pilots in required procedures and maneuvers.

---

[*]Manager, Simulator Development Engineering
Member AIAA

While the FARs "allow" the use of simulators in lieu of airplane training, in reality it is impossible to operate large transport airplanes without access to a simulator. The environmental impact, fuel cost, noise restrictions and other considerations such as quality of training combine to make simulation a necessity.

A significant event occurred in 1988 when FAR 121.409 was amended to require airplane operators to use an approved simulator for each airplane type, in each of its pilot training courses, that provides training in at least the procedures and maneuvers set forth in the certificate holders approved low-altitude windshear flight training program, if their airplanes are required to have windshear detection and guidance equipment installed. This was the first time that training in simulators was required rather than permitted in lieu of airplane training.

An airplane operator may either procure and operate their own simulators, making them user/operators, or use simulators that are operated by another party.

The purpose of this paper is to discuss the role of the simulator user/operator in

obtaining the flight test data required to design and validate the design of a flight simulator and qualify it for use for training and checking of airmen under the previously mentioned FAR parts.

2. Flight Simulator Qualification/Approval Process

Approval for use of a flight simulator in a particular operator's training program will be determined by the Principal Operations Inspector in the case of FAR Parts 63, 121, 125, or 135 certificate holders; or by the Flight Standards District Office responsible for oversight of a training center when the training center is using the simulator to conduct checks required by FAR Part 61. [1] The simulator can be approved only after it has been qualified for approval.

While the FARs allow the use of approved simulators they do not provide a means of certifying their design in a manner similar to that specified for certification of airplane type designs. To certify an airplane, the FAA is involved all through the design and test phases and the result is a certified airplane type design that can be reproduced time after time and the resultant airplane delivered to the customer ready to fly in revenue service. Simulator designs, however, are not certified, rather each simulator that is manufactured is qualified for approval for use on a case by case basis. The responsibility for

qualifying each simulator for each operator is placed with the FAA's National Simulator Program Manager (NSPM).

Under the guidance of the NSPM, advisory circular 120-40, "Airplane Simulator Qualification", has evolved as the primary guidance material for qualifying a simulator for approval. It is the simulator operator's responsibility to apply for FAA qualification, not the simulator manufacturer. This is unlike the airplane certification process, wherein the airplane manufacturer is the certification applicant.

The simulator operator must verify that this particular simulator meets all the requirements of AC120-40 for the level of qualification being sought. The advisory circular specifies the minimum simulator, visual, and motion system requirements for each possible level of qualification, a set of validation tests to objectively compare the performance and handling qualities of the simulator to the airplane using flight test data for comparison, and a table of functional and subjective tests to check that the airplane systems functions have been adequately replicated.

While the airplane manufacturer must meet a large body of regulatory requirements in order to achieve certification for their airplane design, they are under no regulatory requirement to provide any parts, data, or services for qualification of simulators to train the pilots who will fly

their airplanes. Each airplane manufacturer has assumed a position regarding their support for simulator data. Their positions vary widely and range from almost full cooperation to absolute refusal to support simulator data programs. The simulator operator has the most leverage with the airplane manufacturer when they are engaged in negotiations to buy airplanes. The simulator operator should make sure his negotiators are aware of the simulator data requirements and the possible negative effects a lack of data can have in pilot training. Every effort should be made to include a simulator support clause in the airplane purchase agreement that includes, among other items, a simulator data package appropriate to the qualification level intended for the simulator.

In order to build a flight simulator that will meet the performance and handling qualities requirements of AC120-40 the simulator manufacturer must have access to a data package to enable them to implement an aerodynamic model, a ground handling model, a powerplant model, a flying controls model, an autopilot and stability augmentation model etc. and validate these against flight test data.

The simulator operator is the only party that may apply to the NSPM for qualification of the simulator to enable it to be approved for use in an airplane operator's training program. Therefore, it falls upon the simulator operator to be a coordinator between the sources of the airplane data package, the simulator manufacturer, and the FAA. Note that the airplane operator and the simulator operator may or may not be the same party.

The primary role of the user is in the management of the project. The simulator user/operator is the only party in a simulator acquisition and use program present during the entire simulator program, ie. from airplane procurement through pilot training. The simulator data package has implications in all these areas. One aspect of the project is flight testing for the purpose of gathering data to design, validate, and qualify the simulator.

Unfortunately, there are no officially recognized regulatory standards existing like those found in the airframe, engine, and avionics industry that can be used to specify the required data and insure its quality, accuracy, and acceptability for use in the design and validation of a flight simulator. In view of this lack of standards, it is the user's task to monitor all the activities in the simulator project, from the initial airplane procurement through the life of the simulator.

3. Determination of the Flight Test Program Scope

At this time, the most widely recognized guide for definition of the scope and content of a data package necessary to build

commercial flight simulators is, "Flight Simulator Design & Performance Data Requirement", 3rd edition, 1990, published by the International Air Transport Association.

In most cases the user/operator will contract with various vendors including the simulator and visual system manufacturers, the airplane manufacturer, the avionics vendors, and possibly a flight test and model development organization to actually produce the simulator. In some cases, the user will perform some of these activities. When this occurs, the user/operator should be treated like any other vendor by the project management.

Once an operator has reached a decision to procure a simulator or upgrade the qualification of an existing device, whether for use in training their own airmen, or for use in a contract training center, they should determine the scope of the work that will be required to obtain a satisfactory data package for design and qualification of a simulator or flight training device at a particular qualification level. These programs generally fall into one of four categories.

- Existing airplane with an established data package.

- A new airplane type or a significant derivative for which no data package exists, but will be developed by the airplane manufacturer.

- A new airplane type or significant derivative for which no data package exists and the manufacturer will not develop a data package.

- Existing airplane without a data package or a partially complete data package.

The easiest scenario occurs when the simulated airplane falls into the first category. Since the flight test program has been successfully completed and the model and data shown to be satisfactory, the user needs only to obtain the rights to use the data and then insure that the simulator manufacturer correctly implements the models and the data for this simulator.

In the second case the user/operator should encourage the airplane manufacturer to form an ad hoc working group composed of:

- Airframe, engine, avionics, and other significant subsystem vendors.

- The airplane manufacturer's flight test department.

- The airplane manufacturer's simulator model development group.

- Potential simulator vendors.

- Airplane operators obtaining simulators.

- Regulatory representatives.

This group should produce a data requirement to satisfy the simulator modelling needs and to generate data to satisfy the validation requirements of the various regulatory authorities.

Most of the major airframe manufacturers of large transport category airplanes have embraced this type of program for simulator data package development, but it is not universally accepted. Manufacturers of smaller airplanes used by regional airlines, corporate aviation departments, and general aviation operators have not been as ready to accept this type of responsibility.

For a new airplane type or a significant derivative, the most desirable time to generate flight test data for simulators is during the airplane certification flight test program. The user/operator should make every effort to include a simulator data package as part of the airplane purchase agreement or modification program. There are times however when the airplane manufacturer or modification center will not take on this task at any price. This leads to the third and most difficult type of project.

The first requirement of a flight test program is a flight test airplane. If the airplane manufacturer will not participate in the simulator program it will not be possible to test prior to airplane certification, thereby delaying

the pilot training program. The user should make the airplane purchasing group in his organization aware of this fact.

Most simulator manufacturers do not have the capability to do a flight test program and generate the models required for real time simulation. In these cases, a flight test and model development organization must be added to the simulator development team. The user/operator, the simulator manufacturer, and the flight test organization should develop a statement of work to cover all elements of the flight test and model development program. The statement of work used in a recent program included the following items:

- Model architecture definition.

- Review of available data from the airplane manufacturer to determine its acceptability for use.

- Flight test planning responsibilities.

- Flight test instrumentation requirement.

- Flight test requirements for model development.

- Flight test requirement for validation data gathering.

- Model refinement and proof of match requirements.

The statement of work should cover each of these items and expand them to a very detailed level. It is vital that all aspects of the flight test airplane operation be defined. Who provides the airplane, pilots, insurance, maintenance, fuel etc? Where will the flight tests be conducted? How long will the airplane be out of service? What modifications must be accomplished to instrument the airplane? All deliverables from each party to the agreement should be clearly delineated.

The final type of program, data for an existing airplane, is handled the same way if it involves a complete data gathering and model development task. If the job is to fill in the holes in an existing data package the user should be sure that the responsibility for integrating the old and new data is clearly defined. This can be a complex task. For instance, simulators for some of the airplanes that are now candidates for re-engining were built with data packages composed of the best data available at the time. Some of this data was never intended to be used in real time simulation for pilot training and has no proof of match.

It is most important that the simulator operator determine which category a simulator falls under prior to entering an agreement with a simulator manufacturer so that the responsibility for obtaining the data may be properly addressed in the agreement. The user/operator should meet with the airplane manufacturer to

determine into which category this program falls. The data requirements should be clearly stated in the Requests for Proposals (RFP) sent to each potential simulator vendor and then in the specification and contract. There should be a clear understanding of exactly what will be provided, who will provide it, when it will be delivered, and who will pay for it.

The airplane manufacturer's flight test data is the accepted standard for qualification of flight simulators. When the simulation flight test data originates from a source other than the airplane manufacturer, the user/operator should contact the regulatory authority well in advance of the flight test to insure the acceptability of the plans for collecting data.

4. Data Package Content

Regardless of who supplies the data package the content should be the same. The data required by user/operators and simulator manufacturers to design, manufacture, test, qualify and approve a simulator can be categorized as:[2]

- Configuration/Design Data

- Simulation Modelling Data

- Checkout Data

- Flight Test Validation and Proof of Match Data

- System Verification Data

The configuration/design data are those required to design

and fabricate replicas of airplane structure to replicate the airplane cockpit.

The simulation modelling data are those required to define the mathematical construction of realtime simulation of the airplane aerodynamic characteristics and performance of various airplane systems, and the numerical data to support them.

Checkout data are those data intended to verify the correct implementation of the real time models and data used for simulation modelling.

Flight test validation data are the time histories and graphical representations of data obtained from airplane flight test. These data are the resource for comparison with simulator performance for qualification of the simulator by the regulatory agency.

Proof of match data compares the flight test validation data with the performance of the airplane manufacturer's engineering simulation.

5. Flight Test Plan

Regardless of which category the program falls in to, after the available flight test data has been compared to that required in a simulator data package, a detailed flight test plan should be constructed to fill in the holes in the existing data or generate a whole new package. Proper consideration of the following items must be an intrinsic part of the flight test plan:

- Appropriate and sufficient data acquisition equipment.

- Current calibration of data acquisition equipment and airplane installation. (calibration must be traceable to a recognized standard)

- A flight test plan including:

  1. Maneuvers and procedures.

  2. Initial conditions.

  3. Flight condition.

  4. Airplane configuration.

  5. Weight and center of gravity.

  6. Atmospheric ambient and environmental conditions.

  7. Data required from each test.

  8. Other appropriate factors.

- Appropriately qualified flight test personnel.

- Data reduction and analysis methods and techniques.

- Data accuracy.

- Data must be presented in a formats suitable for their intended use ie. design, analysis, or qualification.

- Resolution of data used for qualification must be sufficient to determine compliance with the tolerances.

- Presentation must be clear with necessary guidance provided.

- Overplots must not obscure the reference data.[3]

The flight test plan should also contain provision for a flight test summary report to validate the test results.

Finally, the flight test plan should include a "quick look" capability to enable the flight test engineers, the model developers, the simulator manufacturer, and the simulator user/operator to determine the acceptability of the flight test maneuvers and the data generated by them for use in their particular discipline. If the data are not acceptable or the maneuvers were flown incorrectly then they can be re-flown. This is most critical for flight test programs done solely for collection of simulator data. Almost invariably these programs involve using an airplane removed from revenue service and there is a great deal of pressure to keep the flight test time to a minimum so that the airplane can be returned to revenue service. Once the airplane is de-instrumented, it is too late to correct errors and this can be a very difficult position from which to recover.

6. Post Flight Test Activities

The major phases of a simulator program following flight test are:

- Model development and validation .

- Simulator design, manufacture, and validation.

- Simulator qualification.

- Pilot Training.

Each of these program phases are a subject unto themselves and beyond the scope of this session.

7. Summary

The primary task of the simulator user/operator is to insure that every simulator program is evaluated properly to determine the particular flight test requirements for each program, and provide a plan and program structure to address these needs. The user/operator is the only party involved with the entire span of program activities and therefore in a unique position to provide a structure in which to successfully accomplish the program.

Flight test data is as essential to the success of a simulator program as the airplane parts, instruments, computers and avionics. It should receive as much attention as any of these items.

1.Federal Aviation Administration Advisory Circular, AIRPLANE SIMULATOR QUALIFICATION, AC120-40B, p5, Draft February 25, 1991

2.International Transport Association, Flight Simulator Design & Performance Data Requirements, 3rd Edition, 1990, chapter 1.

3.Federal Aviation Administration, AIRPLANE FLIGHT TRAINING DEVICE QUALIFICATION, AC120-45A, 04/10/90, p7.

FLIGHT TESTING FOR SIMULATOR DATA

F. C. Hall[*]

Boeing Commercial Airplane Group
Seattle, Washington

ABSTRACT

Advances in simulation technology have led to more extensive use of simulators in crew training and engineering analyses. As simulation has become a more prominent part of the design process, flight tests devoted solely to providing data for the aerodynamic data base and crew training simulator validation have become significant elements of an airplane program. This paper describes the content of simulator flight testing for two Boeing models, the 737-500 and 747-400. Maneuvers used to validate the model for crew training simulators and those used to develop the aerodynamic data base are discussed.

INTRODUCTION

Aerodynamic data and math models for crew training simulators in the 1960s were analog implementations. These early simulations were largely based upon wind tunnel data and theory. Flight data to validate were obtained primarily from airworthiness certification and development test flights. Flight testing for these purposes was performed to examine critical flight conditions and critical system performance only. No particular attention was paid to gathering data for typical in-service operating conditions or configurations.

With advances in simulation technology in the 1970s and 1980s, ·a higher level of fidelity

matching simulator to airplane became a reality. First hybrid and then digital simulations became the norm. Regulatory authorities, particularly the Federal Aviation Administration (FAA), developed the Advanced Simulation Plan in 1980. This plan established qualification standards for various levels of training credit and allowed significantly more training on simulators in lieu of airplane flight time. Although the FAA pioneered this approach, other regulatory authorities soon followed so that today many countries have established their own simulator standards. However, most follow the general content of the FAA and United Kingdom Civil Aviation Agency.

What all of these national standards have in common is more evidence that the simulation matches the characteristics of the airplane. It is now necessary to gather flight data to validate the aerodynamic and flight control models for conditions and configurations typical of airline service. The following addresses the flight testing performed on the 737-500 and 747-400 airplanes to provide specific flight data dedicated solely to developing the simulator data base and validating the crew training simulator.

DATA BASE DEVELOPMENT

Two types of testing are required to establish the data base required for simulation. One set of tests is designed for the extraction of aerodynamic coefficients. Some of these tests are a part of the normal development testing typical of a major derivative or new airplane

[*] Supervisor, Stability and Control
Everett Division, BCAG

flight test program. Because
these tests involve configurations
that are not exactly like the
production configuration, they may
not be directly applicable to a
crew training simulator.
Nonetheless, almost every flight
test yields some information that
is useful in establishing the
general aerodynamic data base.

The second set of tests is
comprised of those that are used
to validate a crew training
simulator to the standards at
regulatory agencies. Tables 1 and
2 show the number of tests and
flight test hours for the 737-500
and 747-400 airplanes. About half
of these tests are presented in a
proof-of-match format provided to
the simulator manufacturers and
airlines for developing an
Approval Test Guide (ATG). The
remainder are used for coefficient
extraction and model development.

Although this discussion concerns
the aerodynamic data base, it
should be noted that a similar
process is used for flight control
systems, ground handling, and
propulsion models. These
engineering disciplines perform
comparable analyses and use all
available data to develop the
model and provide proof-of-match
for the ATG.

SIMULATOR-SPECIFIC FLIGHT
TESTING

Flight testing for simulator data
requires special care. Not only
must the maneuvers be performed
precisely and smoothly, the flight
parameters must also be measured
and recorded with an accuracy and
resolution necessary to meet the
standards of regulatory
authorities. Another requirement
that frequently is most difficult
to meet is that of calm
atmospheric conditions with little

or no turbulence. The effect of
winds on the data is particularly
significant for maneuvers of long
duration. Even with precise
inertial data, it remains most
difficult to separate the effects
of the atmosphere from other
dynamic influences.

The procedure also involves
recording several seconds of
hands-off trim prior to each
dynamic or hands-on maneuver.
Even very small levels of
untrimmed accelerations can
contaminate a condition that, if
perfectly trimmed, could be easy
to match. Phugoid response and
flap gear and power change
dynamics are examples of maneuvers
that need both calm air and
accurate trim.

Flight test instrumentation is
another area requiring special
attention when performing
simulator testing. Accurate
measurement of center of gravity
and fuel loading are essential, as
well as the primary parameters of
angle of attack, body attitudes,
control surface position, and
airspeed. In ground effect tests,
height above the ground, and the
influence of ground proximity on
angle of attack and airspeed
measurements are particularly
important.

An extensive array of flight test
instrumentation and recording
apparatus is required.
Instrumentation engineers control
and monitor the parameters that
are recorded on magnetic tape
recorders. Water barrels are used
to adjust the center of gravity
for tests that require special
loading conditions. A trailing
cone is used to measure remote
static pressure.

Many of the preceding
considerations contrast with basic

MANEUVER	NUMBER OF CONDITIONS	FLIGHT TEST HOURS
Stalls	6	0.8
Longitudinal Control	31	3.0
Lateral Control	37	3.9
Trim Characteristics	68	6.5
Dynamic Characteristics	30	2.8
Takeoffs	6	1.2
Ground Effects	26	6.2
Small Control Inputs	30	3.2
Decelerations	35	4.4
Landing Characteristics	8	1.2
Level Flight Acceleration	3	0.4
Static Longitudinal Stability	7	0.7
Reverse Thrust Controllability	5	1.3
Go-Around	2	0.4
TOTAL	294	36.0

Table 1
737-500 SUMMARY

MANEUVER	NUMBER OF CONDITIONS	FLIGHT TEST HOURS
Stalls	25	3.3
Longitudinal Control	0	0
Trim Characteristics	3	0.3
Dynamic Characteristics	43	8.1
Takeoffs	7	0.8
Ground Effects	21	6.9
Small Control Inputs	63	3.8
Decelerations	4	0.6
Landing Characteristics	6	0.7
Level Flight Accel/Decel	10	2.3
Go-Around	3	0.4
Climb	5	2.1
Latency	9	0.3
Flap Failures	8	1.3
TOTAL	240	38.7

Table 2
747-400 SUMMARY

2a

handling qualities airworthiness certification testing which is largely subjective. This subjective nature of much of the certification testing and the fact that airworthiness is examined primarily for critical corners of the flight envelope at limit loading and weight conditions precludes using these data for crew training simulator validation. Although these tests are useful in the development of the complete aerodynamic database, they must be supplemented with simulator-specific tests. The two types of tests mentioned above are addressed in more detail below.

SIMULATOR VALIDATION TESTS
The majority of validation tests recognized today are defined in the current draft of FAA AC 120 40B and CAP 453 of the UK CAA. Regulatory authorities of Australia, France, Japan, and New Zealand also have published simulator validation standards. Currently, a working group sponsored by the Royal Aeronautical Society is drafting a document with the objective of a common international standard acceptable to all ICAO member states. The quantitative tests agreed to so far are based upon AC 120 40B with some tests added by the participating regulatory authorities in the Working Group. Tests fall into three general areas of validation.

1)   Normal Flight Maneuvers:

        Takeoff - normal, crosswind, engine out

        Climb - all engine and engine out

        Stopping - separate effects of wheel brakes, reverse thrust, speedbrakes

        Configuration Changes - flaps, gear, speedbrakes, thrust

        Small Control Inputs - all three axes

        Go around - all engine and engine out

        Landing - normal, crosswind, and engine out

2)   Special Flight Maneuvers:

        Engine out, engine failure

        Minimum unstick (V_{MU})

        Refused takeoff (RTO)

        Stalls

3) Basic Airplane Characteristics:

        Trims

        Longitudinal short period

        Phugoid

        Dutch roll

        Roll response/overshoot

        Static longitudinal stability

        Static lateral-directional stability

        Maneuvering stability

        Level flight acceleration/deceleration

        Ground effect

        Reverse thrust control (symmetric and asymmetric)

For some of these tests, a few words of emphasis are in order. It is recognized that the Normal

Flight Maneuvers are fundamental to most airline crew training curricula. Takeoffs, go-arounds, and landings are prime training maneuvers. Because these are hands-on, relatively long, and conducted largely in ground effect, it is especially important that they are flown in calm air. Crosswind conditions, of course, involve significant atmospheric effects. Winds and the subjective nature of a pilot's evaluation of crosswind operation make these maneuvers particularly difficult to validate with proof-of-match time histories.

Configuration changes on the other hand, when flown in calm air and performed hands off (or free response), are most valuable in developing a math model that accurately represents airplane characteristics.

All Normal Flight Maneuvers should be flown with the airplane in a typical training or airline service configuration. Conversely, Special Flight Maneuvers are conducted generally as part of the development and/or airworthiness certification and involve extremes in landing, gross weight, and flight configuration.

Special Flight Maneuvers may be flown only once in a flight test program, and the resulting data may be all that are available for validation of these simulator characteristics. As these probably are not part of a crew training program, it is important for training simulator evaluators to understand just how these maneuvers are performed in a certification program. There are two exceptions, however: Stalls are believed to be important to be performed as simulator-specific tests at typical landing conditions in addition to the forward center of gravity tests used to verify stall speed and aft center of gravity tests for stall characteristics.

The other exception is for engine failure characteristics at various flight conditions. How the airplane behaves in response to an engine failure is important in crew training. As a general practice, fuel cuts, particularly from high power settings, are performed the fewest possible times required for airworthiness certification. Fuel cuts that produce a rapid thrust decay require special inspections to look for damage and can lead to premature engine removal. Such results are not welcome during a test program and an engine that has been subjected to fuel cuts from high power settings is not always deliverable for airline service without restoration. This is an area where some flexibility on the part of the regulatory authority and ingenuity on the part of the simulator engineer are necessary to reconcile the need for the data with the cost.

Basic Airplane Characteristics are used for math model development as well as for validation of the static and dynamic characteristics of the simulation. For static characteristics, single and three-axis trims must cover the entire operating/configuration envelope of the airplane.

Similarly, dynamic characteristics must be modeled correctly for the complete envelope, although not as extensively as with static trims. The longitudinal short period, phugoid, and Dutch roll characteristics are verified for the unaugmented airplane for various flap and center of gravity combinations. The phugoid and Dutch roll simulations must be compared to flight data for

frequency and damping for several cycles of controls-fixed, free response. As noted previously, these maneuvers are particularly susceptible to poor trim and unsteady atmospheric conditions. For lateral-directional tests it is sometimes necessary to intervene with small column inputs to maintain altitude.

Roll response does not vary significantly with center of gravity but is most sensitive to flap configuration, angle of attack, and airspeed. These maneuvers are tested with yaw damper on and off and for small, intermediate, and full wheel inputs.

Static longitudinal (speed) stability data are generally taken from certification data and represent critical airworthiness test conditions. Wings level or constant heading sideslips are performed to evaluate static lateral-directional characteristics. Sideslips in takeoff and landing configurations are of most interest and where configurations important for crew training have not been obtained in development or certification testing, they must be added to the simulator-unique test program. Sideslips are also valuable in assessing both lateral and directional control.

Ground effect tests are especially critical. A number of techniques have been tried on various airplane programs and some conclusions can be drawn from this experience. First, the data are extremely sensitive to turbulence and the accurate measurement of angle of attack and airspeed data as a function of height above the ground. Level flight fly-bys at various heights are perhaps of value in validating the ground

effect model but have been found to be of limited use in the development of the model. A maneuver that yielded good data for the 737-500 was the "hands-off" landing initiated from a shallow approach (approximately one degree flight path angle). No column input was used between approximately 100 feet AGL and 20 feet AGL. A gentle flare was then used to arrest the sink rate prior to touchdown.
Tests such as these are ideally planned for a location such as Edwards AFB where the surface under the flight path is as level and regular as possible, making radio altitude measurement extremely accurate. Additionally, these tests are conducted near sunrise to take advantage of the temperature inversion and associated calm air.

It should be noted that the cost of ground effect tests should not be borne totally by the crew training simulator test program. These data are also required for the development of control laws for automatic landing systems and their certification. This is true for some other testing as well (see Aerodynamic Data Base Tests, below).

Finally, reverse thrust control is determined for both symmetric and asymmetric reverse thrust conditions. These tests are typically performed with a castoring nosewheel so that aerodynamic control alone is defined. The most important result is the determination of the speed at which aerodynamic control can no longer maintain heading and either symmetric reverse thrust must be restored or reverse thrust cut back to maintain directional control. Development of the model for this characteristic is complex and its fidelity involves strong

influence of speed, level of reverse thrust, and the presence of even low levels of crosswind that exist at the time of the test. Also, no direct measure of reverse thrust magnitude is made and the gross effect measured during deceleration tests must be divided between the direct reverse thrust component and aerodynamic interference. Again, directional control with reverse thrust is a certification test and such data may be all that are available unless specifically defined for simulator development/validation.

AERODYNAMIC DATA BASE TESTS
Aerodynamic coefficients extracted from flight data are used to update the preflight math model and data. The extraction method used involves driving the preflight simulation to match the accelerations measured in flight and comparing aerodynamic parameters in the simulation with those derived from flight test measurements. The key to this process is to minimize the number of independent variables that vary during any particular maneuver. Additional maneuvers have been designed and flown for the sole purpose of providing data for coefficient extraction. These include:

    Trims

    Deceleration

    Roll rate reversals

    Ground effect flares

    Takeoffs - no configuration change

    Constant attitude takeoffs

    Shallow approach with sideslip

Trims are valuable in determining the effects of center of gravity, symmetric and asymmetric thrust, flap setting, Mach number, and aeroelasticity. Many trims must be performed because only one independent variable can be examined for each trim and the complete operating/configuration envelope needs to be covered. Trims are also used in conjunction with decelerations.

Decelerations are typically performed by setting up a trim for a specific configuration, reducing the thrust to idle, and maintaining a constant, low rate (less than 1 knot/sec.) deceleration until stick shaker is reached. This allows a sweep through a large range of angle of attacks in a relatively short period of time (as compared to the large number of trims that would be required). Decelerations are used to obtain basic lift, drag, and pitching moment. In addition, they are used to extract the incremental effects of center of gravity, thrust, flap setting, speedbrakes, and landing gear.

Roll rate reversals are designed to examine the rolling moment coefficient at a point where there is no input due to roll rate. The maneuver begins in a 30 degree banked turn. The pilot is instructed to target wings level using approximately 1/3 wheel. About 5 degrees of bank before wings level, the pilot reverses the wheel deflection to give zero roll rate at wings level. At this point, there will be a specified wheel position in the opposite direction and a substantial roll acceleration, but roll rate will be zero. The rolling moment coefficient for the given wheel input is simply a function of the inertia and roll acceleration.

Ground effect flares have proven to be very effective in updating the ground effect model. Much more data are derived from this maneuver when compared to that from level flight flybys. These tests are used to extract the aerodynamic coefficients in ground effect over a range of angles of attack. The maneuver begins with a shallow approach at constant thrust. The pilot is then instructed to reduce thrust, flare, and maintain a gear height of 2-5 feet above the runway until near the tail contact altitude. This yields data that can be compared to decelerations at altitude (or free air flares).

Special takeoff maneuvers are also used for coefficient extraction for the development of the 737-500 ground effect model. Normal takeoffs are performed with the exception that no configuration change and only minimum control inputs are used after rotation until the aircraft reaches 500 feet AGL. This procedure removes the added complexity of these effects in the ground effect altitude region. Constant-attitude takeoffs are also used to examine the effect of pitch attitude and thrust levels in the ground effect region. Combinations of pitch attitudes of 5 and 12 degrees and thrust levels of 100% MTO and 60% MTO were used for the 737-500 program. These maneuvers proved very useful during development of the takeoff aerodynamic model.

A unique test was performed during the 747-400 program that is still under study. The shallow approach and low altitude deceleration maneuver was performed while holding a small sideslip angle with rudder and wheel. There was an insufficient number of runs to

derive a complete matrix of flap setting, airspeed, and angle of sideslip, however. This is a study item to be pursued in the continuing effort to improve the landing model.

CONCLUSION

For these two programs, both major derivatives of an existing model, the total flight times dedicated to simulation came out to be very close. The distribution of time to specific maneuvers and characteristics, of course, varied for the two airplanes, but overall the order of 35 hours were required to provide the necessary data.

Prior to the 747 testing, the pilots assigned to the simulator data flights were briefed and practiced many of the maneuvers on the engineering simulator. This was a benefit in planning the test sequences, as well as giving the pilots a clear understanding of the test objectives.

It is anticipated that refinements to the procedures used in these programs will lead to more efficient use of the dedicated flight hours. Unfortunately, the most unpredictable feature of such an endeavor is the weather. Measuring and accounting for the effect of wind remains as an important area of study.

Many challenges remain for the simulation engineer. Improved analysis techniques and automation of coefficient extraction procedures are possible with advances in computing hardware and software. Continuous improvement of math models and data quality are keys to both the creation of zero flight time simulators and safer, more productive airplanes.

FLIGHT TEST REQUIREMENTS FOR ADVANCED SIMULATOR DESIGN AND VALIDATION

D. L. Graham
Ferranti Simulation & Training
Orange Park, Florida

Abstract

This paper discusses the quality and quantity of measurements, and the types of test maneuvers required for gathering validation and design data to meet the specific data requirements of flight simulators. Flight test programs of this kind may be conducted by airframe manufacturers, simulator manufacturers, end-user aircraft operators, or independent contractors. A review of existing and proposed U.S. and international regulatory standards is included. The benefits of generating aerodynamic, propulsion, and flight controls design data totally from flights tests are explored. Summary recommendations are made for the design and specification of flight test plans for simulator data acquisition.

Introduction

Modern aircraft flight simulators are designed to replicate aircraft to a very high degree of fidelity. "Zero Flight Time" programs -- where all of the specific aircraft training is accomplished in an advanced flight simulator -- are increasingly common in airline pilot training. Repeatable, verifiable, objective methods are required to validate the fidelity and training transfer of high performance simulators.

Simulator Validation by Comparison to Flight Test Data

The method commonly used to provide for objective simulator validation is to compare the performance and handling qualities of the simulator to parametric data gathered from a representative aircraft in instrumented flight test.

Ultimately, the training objectives of the overall training system should be used to design the specific maneuvers and parameters that will be used to validate the simulator. In the international airline flight training community, the validation of simulators is performed under the guidance of national regulatory authorities.

A flight test program specifically targeted towards the acquisition of simulator validation and design data will be required. In general, only a small percentage of the required simulator data can be expected to be included in the flight test data acquired for purposes of aircraft certification and engineering design. Differences between certification/development flight test and simulator flight test will become obvious as specific maneuvers and parameters are discussed, but can be generally summarized:

• Different weight and CG loadings: much of the simulator data is collected in the nominal "center" of the aircraft envelope; most other flight test is conducted near the extremes of the envelope;

• Different parameter lists: many parameters required for simulator validation (such as detailed control forces and positions, etc.) are not acquired to high tolerances and sample rates for aircraft development;

- Different methods of data observation: nearly all simulator data requires a time history at high sample rates; "snapshots" and other methods are adequate for many aircraft development requirements;

- Different maneuvers: many of the simulator validation maneuvers mandated by regulatory authorities are not required in any other mandated flight test programs.

Regulatory Authority Influence

In 1980, the U.S. FAA implemented the Advanced Simulator Plan. The new, advanced credit training available in simulators was accompanied by advisory guidance detailing the objective requirement to validate advanced simulators against flight test data. The FAA regulatory guidance has been continuously upgraded and updated by inputs from airline users and the simulator manufacturing industry. An excerpt of the current FAA Advisory Circular for simulator validation is shown below as Figure 1.

Similar programs and regulatory guidance have been established by national authorities around the world. Canada, the United Kingdom, France, Germany, Japan, and Australia, among others, have national documents for simulator testing guidance.

Flight test programs intended to cover all validation data requirements for all the individual national regulatory authorities will, today, require reference to each individual document and compilation of a maneuvers list to include them all. In an effort to provide for a consolidated international standard, the Royal Aeronautical Society has sponsored a convention and working group to draft an International Standard acceptable to all of the simulator regulatory authorities. This effort has been ongoing for more than a year, and is expected to provide a draft handbook for potential adoption by the ICAO and JAA in early 1992. An excerpt of the (draft) international document is shown as Figure 3.

The published regulatory guidance against which a civil simulator will be recurrently evaluated, then, is a starting point for defining the specific requirements of a flight test program. However, the regulatory guidance was developed to serve as a "spot check" of the simulator's fidelity. The regulatory guidance assumes that the airline and the simulator manufacturer have conducted comprehensive acceptance tests of the simulator. The specific published maneuvers and test points required by the FAA (or other regulatory authority) Acceptance Test Guide (ATG) should be viewed as a small subset of the maneuvers and test points required to thoroughly evaluate a simulator.

Figure 1 - Excerpt from U.S. FAA Advisory Circular 120-40B

An excellent guidance document for a comprehensive "superset" of maneuvers for simulator validation flight test has been developed and published by the International Air Transport Association: "Design Data Requirements for Flight Simulators", 3rd edition.

Flight Test Maneuvers List

The applicable regulatory authority list of required maneuvers forms the starting point in defining the required validation data flight test program. Specific maneuvers will be required in the following areas:

- Performance
- Handling Qualities
- Takeoff & Landing
- Flight Controls / Control Dynamics
- Free response modes

In addition to the validation maneuvers specifically required by the regulatory guidance, additional data requirements will be determined by the method used to generate the aerodynamic model used in the simulation.

A comprehensive list of the required validation maneuvers from the regulatory guidance is not, however, a complete description of the maneuvers required for a simulator flight test program. The maneuvers list requires expansion into a list of test points that include each maneuver, altitude, airspeed, configuration, gross weight, and other variables.

Selection of actual test points for each maneuver is the prerogative of the organization specifying the flight test program. In the "spot check" nature of FAA (or other regulatory authority) validation, each maneuver will typically only be presented in the FAA Acceptance Test Guide at a single test point and configuration.

The primary purpose for the entire flight test effort is to support the validation of the simulator model against aircraft recorded time histories. The matching of the model against the aircraft will be used to generate and/or refine the model. Consequently, a careful selection of combinations of airspeed, altitude, and configuration will be required to ensure adequate coverage of the training flight envelope. Typically, each "up and away" maneuver will be flown for at least four combinations of airspeeds and altitudes. Each takeoff, approach, and landing maneuver will be flown in multiple configurations and gross weights.

Flight Test Parameter List

The required parameters for a simulator flight test program include as the highest priority those parameters for which a "match" to within specified tolerances is required in the regulatory guidance. However, nearly every regulatory authority requires a large body of supporting data, using language like that excerpted from FAA AC 120-40B:

"Flight test data...may require engineering judgment when making assessments of simulator validity. Such judgment must not be limited to a single parameter. All relevant parameters related to a given maneuver or flight condition must be provided to allow overall interpretation...When comparing the parameters listed to those of the airplane, sufficient data must also be provided to verify the correct flight condition. For example, to show that control force is within +/- 5 pounds in a static stability test, data to show the correct airspeed, power, thrust or torque, airplane configuration, altitude, and other appropriate datum identification parameters should also be given"

Data Quality

Given a prescribed list of maneuvers, test points, and parameters, considerable care must be expended to ensure that the data acquired is of the quality necessary to support the end objective: the design and validation of models in an advanced flight simulator. Factors directly affecting the flight test data quality include qualification and training of the flight test aircrew, the data acquisition system itself, and the data reduction & processing system.

Selection of the flight test aircrew may involve difficult and "ego-bruising" decisions. Flight testing for advanced simulators is done by a very small, very specifically trained and experienced group. The best choice of pilot(s) and flight test engineers is from among those who have done this specific task before. Pilots whose flight test experience has been limited to production and/or certification flight test will require substantial indoctrination and guidance. If possible, crew coordination and maneuver practice should be conducted in an existing simulator prior to the commencement of the test program.

The airborne data acquisition system will obviously have a direct impact on the quality of the entire program. Specific FAA guidance is relatively limited:

"...necessary test parameters electrically or electronically recorded in an airplane using a calibrated data acquisition system of sufficient resolution and verified as accurate by the company performing the test..."

The system used to process, reduce, correlate, and present the airborne data may be overlooked in the review of the data system. However, the ground processing procedures and environment must be carefully designed to preserve the integrity of the data while cross checking and eliminating obvious biases and errors.

Model Generation from Flight Test Data

When a commercial transport aircraft is developed by the larger aircraft manufacturers, the aerodynamic model used for simulation is developed initially from predictive methods, refined by comparison to wind tunnel data, and eventually "tuned" to match the simulator flight test data.

It is, however, increasingly common to develop a comprehensive simulator model completely from the flight test program used to gather the simulator validation data.

Parameter estimation and other analysis techniques may be used to build up the aerodynamic coefficients solely from dynamic flight test maneuvers. The efficiency of the parameter estimation process may be improved substantially by specifying additional flight test maneuvers designed to stimulate specific responses beneficial to the analysis.

A model developed from the simulator flight test data is "self-consistent"; this process eliminates many of the problems encountered in validating a model built up from various sources against the simulator flight test results. The self-consistent model requires even more emphasis on development of test points to provide the widest possible coverage of the aircraft envelope and configurations.

Summary

This paper discusses some of the considerations in designing and developing aircraft flight test programs for the purpose of supporting the design and validation of advanced simulators. Detailed specification of a simulator flight test program includes:

- Development of a maneuvers list, to include all the maneuvers required by national and/or international standards for simulator validation;

- Development of a test point matrix, specifying altitudes, airspeeds, and configurations for each flight test maneuver; the test point matrix requires careful planning to provide maximum coverage of the training envelope while maintaining reasonableness of the time and money required for the flight test program.

- Development of a parameter list that includes all the primary and reference parameters necessary to fully utilize the data developed; and

- Selection, briefing, and training of qualified flight test and data analysis personnel.

FLIGHT TEST DATA FOR SIMULATOR VALIDATION - AN FAA PERSPECTIVE

R. C. Padgett - Hilton G. Smith*
National Simulator Program
Federal Aviation Administration

ABSTRACT

As simulator technology has progressed, the complexity of aircraft types simulated has increased, and the permitted use of simulators has expanded, the need for high quality flight test data has been increasingly emphasized. It has long been the policy of the Federal Aviation Administration (FAA) to require aircraft manufacturer's data to be used to the maximum extent practical for simulator validation, based in part on the concept that the manufacturer has the greatest understanding of its aircraft and can best identify data which defines it. However, data which is more than adequate for certification or performance definition for aircraft is not necessarily of the required quality for accurate simulator validation. Data for simulator validation must be of high quality, repeatable, collected in well-defined conditions with calibrated instrumentation, and presented in acceptable form. Often, in practice, this has not been achieved even when data has been collected specifically for simulation support. High quality flight test data is needed throughout the aircraft's operating range to assure a high confidence level in the performance of the simulator over the entire range of simulator use. This need was evident in the discussions leading to the proposed revision B to the FAA Advisory Circular (AC) 120-40, _Airplane Simulator Qualification_, and is reflected in the changes in tests and test tolerances in the AC. Based upon the experiences of the FAA in the evaluation of simulators, the acquisition of validation data from flight test needs improvement in the conduct, reporting, documentation and presentation of the data.

INTRODUCTION

Simulator technology has advanced tremendously since the first simulator qualification criteria was issued by the FAA in 1965[1]. Computer capability relative to cost has progressed at a rate far ahead of expectations, allowing rapid increases in simulation fidelity, complexity, and capability. This progress

has allowed advanced simulators to be used more extensively in training and airman assessment programs with much less training and checking being done in the aircraft. Commuter and regional as well as business aircraft simulators are also being used more extensively for training and checking. This level of simulator usage combined with the growing complexity of new aircraft emphasizes the need for high quality flight test data for the design and validation of flight simulators. However, in spite of these continued improvements in technology and increased utilization of aircraft simulation, aircraft data for simulator use is often not acquired by the aircraft manufacturer, not of the desired quality nor presented in a form most suitable for simulator validation.

The purpose of this paper is to discuss the requirements and need for improved quality flight test data for simulator validation and give some guidance for the acquisition of that flight test data that will comply with the requirements of References 6 and 7 in the evaluation of advanced simulators.

DATA QUALITY

The requirements for aerodynamic data accuracy and fidelity for the evaluation of advanced simulators are greater than almost any other application in the aerospace industry, including aircraft certification. Final design data are based on predicted/wind tunnel aerodynamic data. Autopilot and stability augmentation systems can be designed and built to less accurate aerodynamic data because prototypes of the "electronic boxes" are usually manufactured to be finely tuned to the exact aerodynamic characteristics of the particular aircraft in flight test. Aircraft certification flight test data often demonstrates that the aircraft meets some particular characteristic rather than define that characteristic. For example, an aircraft must only demonstrate, for certification, that it possesses a minimum positive longitudinal static stability about a trim point rather than define the value of that stability. A qualified simulator, on the other hand, must duplicate the same level of static longitudinal stability as the aircraft. Therefore, all aircraft

certification flight test data are not necessarily acceptable for simulator design and evaluation. Only that flight test data obtained using normal flight test standards and procedures where a sufficient number of parameters including all pertinent aircraft configuration, trim, flight and atmospheric conditions are recorded and properly documented will suffice for these purposes. A simulator must replicate every performance and stability and control characteristic of the aircraft. Validation of this fact requires that these particular characteristics of the aircraft be determined accurately and that the simulator be evaluated by comparison with these aircraft data.

Recognizing the cost and difficulty of acquiring flight test data that meets these requirements for simulator validation, the FAA position has been, since the issuance of Ref. (1), and in all of the guidance issued on simulator qualification, that the airframe manufacturer should be the primary source of these data. This position is based on the concept that the manufacturer has the greatest familiarity with its aircraft and can best identify representative data which accurately defines it. The manufacturer usually correlates the flight test data with his predicted or wind tunnel design data in order to prove or improve analytical skills. He may also employ every possible independent method within reason to verify the validity of the final flight test data realizing the liability incurred upon the publication of the data. Further, design loads data are based upon the manufacturer's final predicted or wind tunnel data and any significant differences between these data and the final flight test data must be addressed in terms of airframe structural load limits, limit speeds, center of gravity limits, etc. Clearly, the manufacturer has a vested interest in the quality, fidelity and validity of the flight test data.

Data for simulators should be of the highest quality. It should be repeatable, and collected in well-defined conditions with calibrated instrumentation using industry accepted procedures and highly qualified personnel. Each test run must be started from a fully trimmed, steady state condition with all parameters which can affect the test being known and recorded. Often, in practice, however, this has not been achieved; even when data has been collected with the participation of the simulator manufacturer. Initial

evaluations of simulators by the FAA since 1987 have provided a number of examples of flight test data of questionable quality. The most common are:

a. Lack of essential parameters such as angle-of-attack, sideslip, roll, or even pitch. Comments on simulator validation packages and discrepancies identified during the FAA evaluation have included Vmca tests without heading information or sufficient yaw rate data to accurately evaluate the test, no runway deviation data for Vmcg, and no yaw or sideslip data for rudder response.

b. Poor test procedures and improperly trimmed conditions. There have been numerous cases where small offsets or accelerations must be included in the simulator initial conditions or a small control input is required upon release of the simulator but prior to the test input in order to match the response of the airplane. Often the only data available begins immediately before a test input and the aircraft's actual initial conditions are unknown or, at the least, uncertain. In some cases incorrect or unusual pilot inputs are made in standard tests, or the aircraft is allowed to drift excessively from the required attitude, such as roll during a phugoid test.

c. Testing in unsatisfactory atmospheric conditions. Too often anomalies in the aircraft data or discrepancies between the simulator data and the airplane data are explained by "atmospheric effects" when little or no wind data is available. This has been most evident in otherwise unexplained changes in airspeed, angle-of-attack, altitude, or pitch. The lack of adequate atmospheric data is a problem in many data packages.

Data package information and explanations on initial conditions, test methods, and data acquisition systems, often fall short of that needed to fully support simulator model development and accurate validation of simulator responses. Too often assumptions have to be made to explain what happened during the flight test or additional questions have to be submitted to the manufacturer for clarification. During the period from early 1987 to the present, more than 30 separate data problems were specifically cited and numerous other cases were discussed with the operator or simulator manufacturer's representatives during initial evaluations of simulators for qualification.

Manufacturers also must provide design data where no other data exists, usually when a simulator is needed for initial instructor and crew training for a new aircraft for

which flight data is not yet available. The problem with design data is that it may not represent all of the changes which have been made to the aircraft since the design data was released and it may represent certain simplifications which do not impact on its use as a design tool but may not be accurate enough for simulator use. Unfortunately, this is unknown until actual test data becomes available. However, this risk is minimized by close coordination and cooperation with the manufacturer and in review of preliminary flight test data prior to final approval of initial simulator training.

REQUIREMENTS FOR FLIGHT TEST DATA

To achieve the level of quality needed in the flight test data for simulator support, the following requirements should be followed:

a. The aircraft should be maintained at the trim condition prior to the start of the test for sufficient time to ensure that it has reached stabilized flight.

b. The initial conditions for the trimmed aircraft should be completely specified for each test (i.e., gross weight, center of gravity, static air temperature, indicated airspeed, flap, gear, and stabilizer or trimming surface positions, wind conditions, power setting, etc.).

c. Data recording devices should begin recording several seconds prior to the start of the test.

d. The tests must be conducted in the most stable atmospheric conditions obtainable and with all atmospheric conditions properly noted.

e. All pertinent parameters should be measured, especially angle-of-attack, sideslip, roll, and pitch, as well as the stability axes angular rates, accelerations and control surface positions.

f. Flight tests must be conducted using calibrated flight test instrumentation and data acquisition systems.

g. Qualified personnel familiar with the flight test procedures and objectives should conduct the tests.

Reference 2 provides guidance on flight test data packages, and references 3, 4 and 5 describe many of the standard flight test procedures which can be used

to collect data for simulator validation, taking into consideration the items discussed above.

PRESENTATION OF VALIDATION DATA

A continuing problem in simulator evaluation, especially for some older airplanes, is the format of the aircraft data. The evaluation of the simulator requires comparing its response to that of the aircraft. Many times the resolution and general quality of aircraft flight test plots are such that the data cannot be accurately read within the tolerance of the specified standard. Another aspect of this problem is that some time histories are so noisy that they are difficult to read correctly.

Time history plots should be presented on scales which allow ease of evaluation. Noisy data should be carefully filtered to reduce the noise but preserve the fundamental characteristic of the parameter. It is recommended that validation flight test data be plotted to standard engineering scales (1, 2, or 4 units or multiples of ten thereof per inch or two centimeters) on standard engineering graphs (twenty divisions per inch or two centimeters) such that all specified tolerances listed in reference 6 and 7 are easily readable. All pertinent parameters for each specific test should be plotted, not just the parameters for which a tolerance is specified in reference 6 or 7. Any differences between the test aircraft and the production aircraft should also be provided.

CONCLUSIONS

Advanced simulators for modern, high technology aircraft require accurate, high quality flight test data for their evaluation and qualification. These data are also required for the accurate assessment of simulator fidelity. Data collection for validation of high fidelity simulators requires flight test programs that take into account the evaluation data requirements of advanced simulators. Aircraft certification and performance flight test data as well as existing manufacturers flight test data is often not sufficiently complete or accurate to validate advanced, high fidelity simulators. A flight test program designed to obtain simulator validation data should consist of well-documented, stabilized tests conducted in calm air using recently calibrated flight test instrumentation and data acquisition systems by experienced flight test engineers and pilots. The final data should be presented in acceptable engineering format that affords the necessary accuracy and

resolution to comply with the requirements and tolerances of References 6 and 7.

REFERENCES

1. FAR Part 121, Appendix B, Minimum Standards for the Approval of Airplane Simulators, 1965.

2. IATA, <u>Flight Simulator Design & Performance Data Requirements</u>, 3rd Edition, 1990.

3. FAA, AC 25-7, <u>Flight Test Guide for Certification of Transport Category Airplanes</u>, April 9, 1986.

4. FAA, AC 23-8A, <u>Flight Test Guide for Certification of Part 23 Airplanes</u>, February 9, 1989.

5. FAA, AC 29-2A, <u>Certification of Transportation Category Rotorcraft</u>, September 16, 1987.

6. FAA, AC 120-40A, <u>Airplane Simulator Qualification</u>, July 31, 1986.

7. FAA, AC 120-40B, <u>Airplane Simulator Qualification</u>, Draft.

MODEL ORDER REDUCTION APPLIED TO A HOT-BENCH SIMULATION
OF AN AEROELASTIC WIND-TUNNEL MODEL

Carey S. Buttrill* and Barton J. Bacon**
NASA Langley Research Center
Hampton, Virginia 23665-5225

AIAA-91-2935-CP

Abstract

Simulations of an aeroelastically scaled wind-tunnel model were developed for hot-bench testing of a digital controller. The digital controller provided active flutter-suppression, rolling-maneuver-load alleviation, and plant estimation. To achieve an acceptable time scale for the hot-bench application, the mathematical model of the wind-tunnel model was reduced from 220 states to approximately 130 states while assuring that the required accuracy was preserved for all combinations of 10 inputs and 56 outputs. The reduction was achieved by focussing on a linear, aeroelastic submodel of the full mathematical model and by applying a method based on the internally balanced realization of a dynamic system. The error-bound properties of the internally balanced realization significantly contribute to its utility in the model reduction process. The reduction method and the results achieved are described.

Introduction

Minimizing weight is vital to any successful aircraft design. An aircraft must be flutter-free at some percentage (the safety margin) beyond its operating envelope. Flutter is a dynamic instability that occurs on aircraft at a dynamic pressure and Mach condition wherein the structure begins to extract energy from the fluid flow. Traditional, passive remedies for flutter problems add weight in the form of additional structure or mass ballast. If the flutter-free requirement for an aircraft design can be met by active stabilization using existing control surfaces (at least in the margin beyond the operational envelope), significant weight savings may be possible.

In the recently completed active flexible wing (AFW) program,[1,2,3,4,5] a full-span, free-to-roll, aeroelastically scaled model of an advanced fighter was tested in the Transonic Dynamics Tunnel (TDT) at the Langley Research Center (LaRC). Four tunnel entries have been conducted as part of the AFW program: 1986, 1987, 1989, and 1991. The objective of the 1991 test was to demonstrate active flutter suppression while simultaneously meeting scaled Military-Specification roll performance criteria and alleviating roll-induced loads. Achieving the 1991 objective required rolling the model above the open-loop flutter boundary. The 1989 test objectives allowed flutter suppression and roll performance tests to be done separately.

The flutter mechanism of the AFW model was of the clean-wing, classical variety with a rapid onset. Failure of the active flutter suppression system while testing above open-loop flutter conditions would put both the model and the tunnel at risk. An extensive simulation-based testing and

validation effort was needed for the AFW program. Simulations produced to support the 1989 and 1991 tunnel entries are the subject of reference 6 and this paper.

Both batch and hot-bench simulations were developed for each tunnel entry. The batch simulations served as "truth" models and were used to: (1) evaluate the control laws by predicting performance and establishing gain and phase margins, (2) provide data files for the hot-bench simulations, and (3) verify the hot-bench simulations. The hot-bench simulations of the model in the wind-tunnel environment served to support testing of the digital controller system.

The simulation mathematical model developed for the 1991 tunnel test was too large to provide an effective hot-bench simulation. Substantial reduction was required without compromising response accuracy. The model reduction methodology applied to the 1991 mathematical model was based on the well known properties of the internally balanced (IB) realization of a linear time-invariant system.[7,8] When an asymptotically stable system is transformed into IB coordinates, the resulting states are ordered in their importance to the system's output/input performance. An easily calculated bound exists for determining the error introduced by deleting the least significant IB states. Furthermore, this bound applies to the maximum frequency response error over all frequencies for all possible output/input paths. The IB method was extended by Enns,[9] who developed an approach whereby frequency regions could be selected for emphasis by the use of weighting filters (FWIB). Bacon[10] extended the FWIB approach, producing a frequency-weighted pole-preserving (FWPPIB) method to better accommodate a system with unstable or neutrally stable components. For the work described herein, the classical IB method of handling unstable subsystems by decoupling the stable and unstable portions was applied without frequency weighting. When the classical IB method was combined with appropriate scaling and a conservative choice for an error bound, a relatively "hands-off" reduction method emerged. The success encountered in applying an IB approach to aeroelastic systems of realistic size and complexity is significant in that it offers an additional approach to the research that has been performed in minimizing the impact of unsteady aerodynamic states on the size of aeroelastic systems.[11,12]

Wind-Tunnel Model

Figure 1 is a schematic of the AFW wind-tunnel model. The model is free to roll +/- 145° about the sting axis. To support flutter suppression investigations, destabilizing mass ballast was added to each wingtip so that the model would flutter within the operating envelope of the TDT.[2] The tip ballast serves to lower both the first-bending and the first-torsion elastic mode frequencies with the predominant effect on the first-torsion mode. The result is that the first-torsion

* Research Engineer, Aircraft Guidance and Controls
  Branch, NASA LaRC, Senior Member, AIAA
** Research Engineer, Aircraft Guidance and Controls
  Branch, NASA LaRC

185

and the first-bending elastic modes combine to form the primary flutter mechanism at a lower dynamic pressure than was the case for the original wind-tunnel model (with no tip ballast). The tip ballast can also be rapidly decoupled in pitch from the wingtip by releasing a hydraulic brake. When decoupled, the tip ballast is restrained in pitch by a soft spring. Decoupling the tip ballast proved to be an effective flutter stopper during testing, providing additional safety margin.

As indicated in figure 1, there are 8 control surfaces to be commanded and 40 sensor outputs, which include 8 control surface position measurements, 13 accelerometer outputs, 8 strain gauge outputs, 8 hinge moment measurements, and model pitch position, roll position, and roll rate. Reference 13 describes, in detail, the AFW wind-tunnel model prior to adding the tip ballast.

Simulations

Hot Bench Laboratory

The AFW hot-bench simulation set-up, depicted schematically in figure 2, utilized the central real-time facility at LaRC. The LaRC real-time facility consists of nodes that communicate by means of a 50-megabit-per-second fiber-optic digital-data network. The simulation nodes on the network included two Control Data Cyber 175 computers, engineering control consoles, various aircraft cockpits, graphics computers, and motion-base hardware. For the AFW hot-bench simulation, one Cyber 175 was used to integrate the equations of motion. A Terabit Eagle 1000 graphics computer provided a real-time display. The display presented model pitch and roll, control surface deflections, and total structural deformation of the simulated wind-tunnel model. Communications with the digital controller occurred over analog lines in the same manner as when the controller was connected to the wind-tunnel model at the LaRC TDT.

Digital Controller Apparatus

The control computer consisted of a SUN 3/160 workstation augmented with a digital signal processor, a floating point array processor, and analog/digital conversion boards.[2,3] During both hot-bench and tunnel testing, signals to and from the digital controller go through an interface box. The interface box contained trip logic that tested for a variety of exceedances. If a trip occurred, the control loop would be opened, the tip ballasts would be commanded to their decoupled, stable state and, depending on the test conditions, the roll brake would be engaged. In addition, the interface box housed anti-aliasing filters that conditioned the signals from the wind-tunnel model before being sampled by the digital controller.

In the hot-bench laboratory the 25 Hz single-pole analog anti-aliasing filters in the model/controller interface box were bypassed and their dynamics included in the simulation of the wind-tunnel model. The hot-bench environment is relatively noise-free when compared to the wind-tunnel environment and the absence of anti-aliasing filters in the hot-bench set-up was not felt to be a difficulty. The presence of analog dynamic elements would have interfered with the need, discussed in a subsequent section, to run at a variety of time scales. Real-time simulations supported at the LaRC facility are typically run without anti-aliasing filters unless problems are noted.

In addition to its primary function of implementing control laws at 200 Hz, the digital controller performed a variety of support functions. Static checks, dynamic data acquisition and storage, and plant excitation were performed. Excitation data were shipped to another computer wherein both open-loop and closed-loop plant estimations were performed.[4]

Mathematical Models

An overview of the structure of the batch and hot-bench simulations and the data flow between them is shown in figure 3. There are 10 inputs into the simulation mathematical model (8 actuator commands and 2 Gaussian random numbers used to drive the turbulence models). Fifty-six outputs are generated. The first 40 outputs represent the simulated wind-tunnel model sensor outputs: accelerometers, surface positions, strain gauge outputs, etc. The last 16 outputs are the displacements of the generalized coordinates associated with 16 (2 x first 8 per symmetry) elastic modes and are required to drive the graphics display. The mathematical models of both the batch and hot-bench simulations, the method of discretization used in the hot-bench simulation, and the method whereby the hot-bench simulation is generated automatically from the batch simulation are described in reference 6.

The mathematical model can be viewed as three submodels connected (with the exception of hinge moments) serially. The first submodel consists of the primary first-order poles associated with eight empirically derived actuator transfer functions. Each actuator transfer function consists of a first order term and a complex conjugate pair in the denominator and a constant numerator. The first-order pole reflects the dynamics of hydraulic fluid flowing through a small orifice whose size is regulated by position feedback. The complex conjugate pair is affected by the compressibility of the hydraulic fluid and the position feedback gain. Actuator rate-limiting is imposed in the first submodel.[6] The second submodel consists of the aeroelastic equations together with the remaining second-order portion of the actuator transfer functions. If dynamic pressure is held constant, the aeroelastic submodel is linear time-invariant. The third submodel consists of 40 linear first-order anti-aliasing filter equations applied to the simulated sensor outputs of the second (aeroelastic) submodel.

In preparation for the 1991 test, the mathematical model was updated in an attempt to improve the ability to predict responses observed in the 1989 tests.[6] As a result, the aeroelastic submodel grew from 73 states in the 1989 mathematical model to 172 states in the 1991 mathematical model (table 1). The 1991 mathematical model used a four lag least-squares[11,12] approximation of the unsteady aerodynamics and the resulting unsteady aerodynamic lag states account for the bulk of the increase in state dimension.

Hot-Bench Time Scaling

The AFW hot-bench simulation operated at a time scale slower than 1:1 (real time). Time-scale is a function of the integration step (h) and the real-time computer frame size (T). If T is larger than h, the simulation runs at 1:(T/h) "slow." Simulating the anti-aliasing filters ensured that all dynamic elements in the hot-bench loop were digital. Therefore, the hot-bench loop could be run synchronously at a slow time scale and dynamic validity would be maintained.

Several factors prevented the AFW hot-bench simulation from operating in real time. The control computer runs at 200 Hz in the wind-tunnel. To prevent excessive digitally induced time delays, the hot-bench simulation needed to update twice for each digital controller frame. If the hot-bench simulation were to run in real time, a 400 Hz update rate would therefore be required. Using the discretization methods described in reference 6, the smaller 1989 mathematical model had been implemented in the hot-bench simulation with a frame time of 12.5 millisecond (80 Hz) and an integration step of 2.5 milliseconds (400 Hz), resulting in a 1:5 time scale factor when an entire Cyber 175 was available. Even had the Cyber 175 CPU been able to complete the equations of motion calculations in 2.5 milliseconds (400 Hz), the minimum frame time available on the Cyber 175 due to the real-time operating system architecture was 5 milliseconds (200 Hz).

During the hot-bench testing phase of the AFW program, many research programs competed for LaRC's two production real-time computers. Consequently, the hot-bench simulation often shared a Cyber 175 with another job. With only one half of the Cyber computing power available, the 80 Hz simulation update rate would be further reduced to 40 Hz. Time-scale was therefore required to be an easily adjusted parameter for any dynamic component in the hot-bench loop.

During the course of hot-bench testing prior to the development of the 1991 mathematical model, it became apparent that, from a productivity standpoint, a time-scaling factor of 1:5 (full Cyber) was the maximum reduction allowable for effective hot-bench testing. To run much slower meant that too few data-taking runs could be performed in a 2-to-3 hour hot-bench testing session. To maintain the 1:5 time-scaling achieved with the smaller 1989 mathematical model, model reduction techniques were required.

Reduction Method and Application

Description of the Method

The model reduction method described herein is based on transforming a stable subspace of the full dynamic model to internally balanced coordinates. The method given below builds on work described in reference 9. The major difference between the method presented herein and derivations found in reference 9 is in the calculation of the transformation matrix, **T**, discussed in step 6. More complete treatments of the properties of singular value decompositions, controllability and observability grammians, and Hankel singular values are found in references 8, 9, and 10. The intent of the following is to outline the steps required to implement the method and describe its virtues and limitations.

Consider the linear, time-invariant, possibly unstable dynamic system defined by

$$\begin{Bmatrix} \dot{x} \\ y \end{Bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{Bmatrix} x \\ u \end{Bmatrix} \qquad (1)$$

and referred to throughout this paper by the partitioned real system matrix, $\begin{bmatrix} A & B \\ C & D \end{bmatrix}$.

Step 1

Scale the system inputs and outputs to units of similar significance.

Step 2

Transform the system of equation (1) into coordinates that decouple the stable and unstable portions of the model,

$$\begin{Bmatrix} \dot{x}_u \\ \dot{x}_s \\ y \end{Bmatrix} = \begin{bmatrix} A_u & 0 & B_u \\ 0 & A_s & B_s \\ C_u & C_s & D \end{bmatrix} \begin{Bmatrix} x_u \\ x_s \\ u \end{Bmatrix} \qquad (2)$$

where the eigenvalues of A_s are strictly stable.

Step 3

Calculate the controllability grammian, X, and the observability grammian, Y, of the stable subsystem by solving their associated Lyapunov equations,

$$A_s X + X A_s^T + B_s B_s^T = 0 \qquad (3)$$

$$A_s^T Y + Y A_s + C_s^T C_s = 0 \qquad (4)$$

Step 4

Perform a singular value decomposition on the controllability grammian,

$$U_x \Sigma_x V_x^T = X \qquad (5)$$

where Σ_x is a diagonal matrix of positive real numbers and U_x and V_x are real, square, unitary ($U_x U_x^T = V_x V_x^T = I$) matrices. Since X is both symmetric and positive semi-definite, $V_x = U_x$, and the decomposition (5) can be written as

$$U_x \Sigma_x U_x^T = X \qquad (6)$$

Since Σ_x is diagonal, $\Sigma_x^{1/2}$ will refer to a diagonal matrix whose diagonal elements are the square roots of the diagonal elements of Σ_x.

Step 5

Perform a second singular value decomposition on the real symmetric, positive semi-definite product,

$$U_x \Sigma_x^{1/2} U_x^T Y U_x \Sigma_x^{1/2} U_x^T \qquad (7)$$

to give,

$$U_w \Sigma_w U_w^T = U_x \Sigma_x^{1/2} U_x^T Y U_x \Sigma_x^{1/2} U_x^T \qquad (8)$$

where Σ_w is a diagonal matrix of nonnegative real numbers ordered in decreasing magnitude.

Step 6

A transformation matrix, **T**, is then formed,

$$T = U_x \Sigma_x^{1/2} U_x^T U_w \qquad (9)$$

If n_s is the order of the square matrix A_s and n_{sr} is the order of the desired reduced-order model, then the following partitioning of the transformation matrix, T, and its inverse, T^{-1}, can be defined,

$$[\ T_1\ \ T_2\ ] = T \quad \text{and} \quad \begin{bmatrix} U_1^T \\ U_2^T \end{bmatrix} = T^{-1} \qquad (10)$$

where T_1 is $n_s \times n_{sr}$ and U_1 is $n_s \times n_{sr}$.

It should be noted that the transformation, T, is equivalent to the transformation formed in reference 9. If equation (8) is multiplied on the left by $U_x \Sigma_x^{1/2} U_x^T$ and on the right by $U_x \Sigma_x^{-1/2} U_x^T$, the following result is obtained,

$$T \Sigma_w T^{-1} = XY \qquad (11)$$

Thus, the transformation that is achieved in (10) by singular value decompositions, is the same T that is achieved by an eigenvalue decomposition of the product XY, as shown in equation (11) and as discussed in reference 9.

It should also be noted that unless (A_s, B_s) is a completely controllable pair, the diagonal matrix $\Sigma_x^{1/2}$ will contain one or more zeros on the diagonal, rendering T singular. If T is singular, T^{-1} will not exist and T will not be a transformation matrix. Steps 4 through 6 therefore require that (A_s, B_s) be a completely controllable pair, or equivalently, X be positive definite ($X > 0$), not merely positive semi-definite.

Step 7

When the stable subsystem is transformed according to the matrix, T, the new, internally balanced states are ordered in importance to the frequency response over all output/input pairs. A reduced-order model of the stable subsystem can now be formed by the following equations,

$$\begin{bmatrix} A_{sr} & B_{sr} \\ C_{sr} & D_{sr} \end{bmatrix} = \begin{bmatrix} U_1^T A_s T_1 & U_1^T B_s \\ C_s T_1 & D \end{bmatrix} \qquad (12)$$

Step 8

The matrix A_{sr} will generally be fully populated. Transform the reduced-order stable subsystem, $\begin{bmatrix} A_{sr} & B_{sr} \\ C_{sr} & D \end{bmatrix}$, to a real Jordan form to minimize computations required for implementing the discretized form of the reduced system. The real Jordan form is diagonal up to 2×2 blocks for independent eigenvectors. The near diagonal reduced-order stable subsystem can then be recombined with the unstable subsystem of the original dynamic system to form a reduced-order form of the original, possibly unstable, dynamic system,

$$\begin{bmatrix} A_r & B_r \\ C_r & D^r \end{bmatrix} = \begin{bmatrix} A_u & 0 & B_u \\ 0 & A_{sr} & B_{sr} \\ C_u & C_{sr} & D \end{bmatrix}. \qquad (13)$$

Step 9

Scale the reduced system back to original units.

Numerical Robustness

Prior to implementing the procedures described above, a method was first attempted wherein a commercial package was invoked to provide the internally balanced representation of a stable system achieved by the transformation, T, in step 6. It is not known what method the commercial package used to achieve an internally balanced representation, but the command consistently failed and reported a problem with performing a Choleski decomposition. It should be noted that this same commercial package worked fine for smaller systems. The authors believe that the use of singular value decompositions, well known for their numerical stability, make the approach outlined in steps 1 through 9 attractive.

Error Properties of the Method

Having formed a reduced-order model, it is important to know how well the full-order model is approximated. Define a frequency response error matrix E_{sr} by differencing the full and reduced system

$$E_{sr}(j\omega) = C_s(j\omega I - A_s)^{-1} B_s - C_{sr}(j\omega I - A_{sr})^{-1} B_{sr} \qquad (12)$$

Let the n_s diagonal elements of $\Sigma_w^{1/2}$ defined in (8) be denoted by $\sigma_h(i)$ (the Hankel singular values[8]). A magnitude bound[9,10] on the error matrix, E_{sr}, is given by,

$$\sup_\omega \bar{\sigma}[E_{sr}(j\omega)] \leq e_B = 2 \sum_{i=n_{sr}+1}^{n_s} \sigma_h(i) \qquad (13)$$

so that the maximum singular value of the error matrix over all frequencies is less than or equal to twice the sum of the Hankel singular values associated with the removed n_s-n_{sr} states. The maximum singular value of a matrix can be interpreted as,

$$\bar{\sigma}(E_{sr}(j\omega)) = \sup_{|u|=1} |E_{sr}(j\omega)u(j\omega)|. \qquad (14)$$

Thus, the maximum magnitude response of E_{sr} in any direction for a harmonic input of unit size is bounded by e_B. In particular, the magnitude of the largest element of $E_{sr}(j\omega)$ is bounded by e_B. The significance of this well known result for internally balanced realizations cannot be overstated. Once the labor of steps 1 through 5 are performed, the quality of the reduced order model to be generated for any selection of n_{sr} can be determined prior to performing steps 6 through 9. Furthermore, since the stable and unstable portions have been decoupled by (2), the result of (13) holds for the error matrix associated with the complete system,

$$E_r(j\omega) = C(j\omega I - A)^{-1}B - C_r(j\omega I - A_r)^{-1}B_r. \quad (15)$$

<u>Application of the Method</u>

In the 1991 mathematical model, the 172 state aeroelastic submodel constitutes the bulk of the required computations in the equations of motion. When the aeroelastic submodel is written in matrix form as in equation (1), the **A**, **B**, and **C** matrices are highly populated. Even if the system were transformed to coordinates where **A** was diagonal, full matrix multiplications would still be required for **B** (172×10) and **C** (56×172). On the other hand, the computational load of implementing a discretized form of (1) for the third submodel (the anti-aliasing filters) is small when compared to the aeroelastic submodel. The 40 anti-aliasing filter equations are uncoupled. When written in matrix form, the **A**, **B**, and **C** matrices are square and diagonal and the matrix **D** is zero. The aeroelastic submodel was therefore chosen for application of the model reduction method.

Maintaining the 1:5 time scaling required that the aeroelastic submodel, with 172 states, 10 inputs and 56 outputs, be reduced to a system with approximately 80 states that would accurately replicate the full-order output/input results for all 560 combinations. As described in reference 6 and depicted in figure 3, the hot-bench simulation is derived by extracting linear aeroelastic submodels from the batch simulation that are valid for a particular dynamic pressure. To provide sufficient continuity for hot-bench testing, linear aeroelastic submodels were extracted and reduced for 9 different dynamic pressures, ranging from 150 psf to 350 psf in increments of 25 psf. This process was repeated for two wind-tunnel model test configurations (and by extension, for the batch simulation). The two configurations were: (1) the model fixed (roll brake engaged) and (2) the model free-to-roll.

A requirement of any reduction process is the scaling of the inputs and outputs to units of similar significance. The first 8 inputs (actuator commands) and first 40 outputs (sensors) of the simulation correspond to analog lines on the wind-tunnel model. For these inputs and outputs, a natural selection for units was volts. The two turbulence inputs were scaled so that an intensity of one standard deviation was weighted the same as 1 volt. The 16 outputs representing elastic mode deflection were left unchanged.

The analog to digital (A/D) converters in the digital controller yield 12 bits of resolution for inputs with a range of $+/-10$ volts. If an error bound of $e_B=20/2^{12}$ had been satisfied, then for a harmonic signal into any input, the difference between the full mathematical model and the reduced model at any output would have been less than the resolution of the converters. For the scaling described above, error bounds tighter than $20/2^{10}$ produced models too large for the hot-bench timing budget. Since the error bound represents a worst case error and differences in the last two bits of a converted signal would be small compared to electrical noise in the wind-tunnel and inaccuracies in the mathematical model, an error bound of $e_B=20/2^{10}$ was used. This choice for an error bound was validated by time history comparisons between the batch and the resulting hot-bench simulation. Once an error bound is determined, the order of the candidate reduced model, n_{sr}, was varied until the required accuracy was achieved according to the definition of

equation (13). This ensured that the reduced-order aeroelastic submodel would retain 10 of the 12 bits of precision available through the A/D converters. For each of the reduced-order models formed at 9 dynamic pressures, no more than 82 states were required.

Step 2 was implemented by transforming the system of equation (1) to real Jordan form. For the dynamic systems considered in this study, the eigenvectors were independent, resulting in real Jordan forms wherein all non-zero elements of the transformed A matrix are either real 1×1 ($[\sigma]$) or real $2\times2 \left(\begin{bmatrix} \sigma & \omega \\ -\omega & \sigma \end{bmatrix}\right)$ blocks on the diagonal. The 2×2 blocks correspond to system eigenvalues that are complex conjugate pairs, with σ being the real part and ω being the imaginary part. Corresponding rows and columns of the resulting system matrix can then be permuted, taking care to preserve the integrity of the complex conjugate pairs, until all the blocks corresponding to unstable eigenvalues ($\sigma \geq 0$) are in the upper left-hand corner of the system matrix. The result is a system of the form where the eigenvalues of A_s are strictly stable ($\sigma < 0$).

Results of Reduction

Frequency response comparisons between the full-order (172 states) linear aeroelastic submodel and a reduced-order form for selected output/input combinations are shown in figures 4, 5, and 6. The responses for figures 4, 5, and 6 are for the case of the simulated wind-tunnel model being free to roll and at a dynamic pressure of 300 psf. This dynamic pressure is well past predicted instability and ensures that the portion of the reduction method that preserves the unstable dynamics is demonstrated.

Figure 4 shows comparisons of full and reduced models for the case of the transfer function consisting of the right wingtip accelerometer response to the right trailing edge actuator command. The response of the 79 ($e_B=20/2^{10}$) state model is indistinguishable from the full model response for the frequency range shown. Responses for 57 state ($e_B=20/2^3$) and 55 state ($e_B=20/2^2$) models are also shown. For the output/input pair shown of figure 4, the reduction process must be applied fairly aggressively before the reduced-model response departs from the full-order model. The property of an IB-based reduction method to match peak frequency response is clearly demonstrated.

Figure 5 shows the response of same right tip accelerometer to a symmetric turbulence input. In this case the 79 state model does depart somewhat from the full model and one need only go to the 60 state ($e_B=20/2^5$) model to produce substantial departure. The response of figure 4 appears to be fitted better by the 79 state model because the peak response of figure 5 is 20 dB (a factor of 10) higher than the peak response of figure 5. Note that the 79 state model is exact down to -40 db for both figures. If the symmetric turbulence random number input had been scaled in step 1 to units 10 times larger, figure 5 would resemble figure 4.

Figure 6 shows the frequency and time responses of left leading edge outboard control position due to a command at the right trailing edge outboard actuator. The zero response of the full-order (172 state) linear aeroelastic submodel for the

is shown by the heavy solid line. Responses that are identically zero in the full model are allowed to become non-zero in the reduced models. Actuator interaction is modeled only to the extent that calculated hinge moments affect applied rate limits. Thus, in the linear aeroelastic submodel, there is no cross feed between actuators - activity on one actuator does not induce activity in another actuator. The magnitude frequency response for the full-order model is actually zero, but is set arbitrarily at -100 dB by the plotting software.

Figure 7 is a plot of the error bound, e_B, of equation (13) versus number of states retained, n_{sr}, of equation (10). The error will go to zero when n_{sr} reaches 170, as this is the size of the stable component of the aeroelastic submodel at a dynamic pressure of 300 psf.

Also of interest is the end-to-end time response comparison between the entire 220 state batch simulation and the complete hot-bench simulation model. Responses of both batch and hot-bench simulations were calculated for a 1 volt (2.67°) step command to the right trailing edge outboard actuator for the free-to-roll condition at a dynamic pressure of 225 psf. The time response agreement was uniformly excellent for all outputs. A typical result, the right and left tip accelerometers, is shown in figure 9. The solid lines are the batch simulation responses and the dotted lines are the hot-bench simulation responses.

More advanced model reduction techniques employ frequency weighting filters to discount selected frequency regions of the full model.[9,10] In a frequency weighting approach, weighting filters would be applied to either all the input signals or all the output signals. While lower order final models can be achieved by bounding the frequency range of interest, the application of frequency weighting raises three concerns. Foremost is the absence of a guaranteed bound of the form of equation (13). The presence of this bound for the IB method obviates the need for extensive checking of all possible output/input combinations in the reduced models. A second concern is that adding weighting filters imposes a computational and numerical robustness burden by effectively raising the order of the full plant to be reduced. Solving the Lyapunov equation (3) for a symmetric controllability grammian, X, of dimension 100×100 is analogous to solving a linear system of equations for $5050 = \frac{(101)(100)}{2}$ unknowns. Raising the order of X by 10 raises the order of the associated Lyapunov solution from 5050 to 6105. The third issue concerns time responses. Since the internally balanced approach ensures a good match in peak frequency response over the entire frequency range, equivalent responses in both the frequency and time domains between the full and reduced order models are ensured for arbitrary input. If a reduced order plant is produced by de-emphasizing selected frequency components, a step input, which has broad band frequency content, would produce a different time response from the full-order model. Band limited inputs will be required to produce matching time histories. This property should not be a concern as long as it is anticipated and understood.

The classical IB approach in combination with the numerically robust method of calculation presented herein and the system scaling chosen, was well suited to the problem of reducing an aeroelastic simulation model. The simulation model had to give results indistinguishable from the batch simulation to support validation testing of the digital controller. The large number of states, inputs, and outputs would have made the application of frequency-weighted methods difficult and, for the target model size required, unnecessary.

Concluding Remarks

Modern model reduction techniques were successfully applied to a time-scaled hot-bench simulation of an aeroelastic system. The model reduction method is based on transforming a stable subsystem to internally balanced coordinates. The internally balanced approach was particularly effective at removing the aerodynamic lag states that are often part of linear aeroelastic systems.

Since the internally balanced approach ensures a good match over the entire frequency range, equivalent responses in both the frequency and time domains between the full and reduced order models are ensured. The equivalent time and frequency responses significantly simplified the hot-bench validation process.

Finally, the error bound properties of the internally balanced decomposition greatly enhance its usefulness in model reduction. The need for extensive checking of all possible output/input combinations in the reduced models is obviated. However, numerical conditioning can be an issue when trying to perform internally balanced decompositions of large aeroelastic systems and caution is recommended.

References

1. Noll, T.; et al.: *Aeroservoelastic Wind-Tunnel Investigations Using the Active Flexible Wing Model - Status and Recent Accomplishments.* NASA TM-101570 and AIAA Paper 89-1168. Presented at the AIAA/ASME/ASCE/AHS 30th Structures, Structural Dynamics, and Materials (SDM) Conference in Mobile, Alabama, April 1989.

2. Perry, B.; et al.: *Digital-Flutter-Suppression-System Investigations for the Active Flexible Wing Wind-Tunnel Model.* NASA TM-102618 and AIAA Paper 90-1074. Presented at the AIAA/ASME/ASCE/-AHS 31st Structures, Structural Dynamics, and Materials (SDM) Conference in Long Beach, California, April 1990.

3. *Aircraft Active Control Applications*, a session of six papers on the Active Flexible Wing Test Program presented at the Fourth NASA Workshop on Computational Control of Flexible Aerospace Systems held in Williamsburg, Virginia, July 11-13, 1990. NASA CP-10065, pp 459-653.

4. Pototzky, T.; et al.: *Development and Testing of Methodology for Evaluating the Performance of Multi-Input/Multi-Output Digital Control Systems.* AIAA Paper 90-3501. Presented at the AIAA Guidance, Navigation, and Control Conference in Portland, Oregon, August 20-22, 1990.

5. Waszak, M. and Srinathkumar, K.: *Active Flutter Suppression: Control System Design and Experimental Validation.* AIAA Paper 91-2629-CP. Presented at the AIAA Guidance, Navigation, and Controls Conference

in New Orleans, Louisiana, August 12-14, 1991.

6. Buttrill, C. and Houck, J.: *Hot-Bench Simulation of the Active Flexible Wing Wind-Tunnel Model*. NASA TM-102758 and AIAA Paper 90-3121-CP. Presented at the AIAA Flight Simulation Technologies Conference in Dayton, Ohio, September 17-19, 1990.

7. Moore, B.C.: "Principal Component Analysis in Linear Systems: Controllability, Observability, and Model Reduction", *IEEE Transactions on Automatic Control*, VOL. AC-26, NO 1, February 1981, pp 17-32.

8. Glover, K.: "All Optimal Hankel-Norm Approximations of Linear Multivariable Systems and their L∞-Error Bounds", *International Journal of Control*, 1984, Vol. 39, No. 6., pp.1115-1193.

9. Enns, D.F.: *Model Reduction for Control System Design*, Ph.D. Dissertation, Dept. of Aeronautics and Astronautics, Stanford Univ., Stanford, California, June 1984.

10. Bacon, B.J.: *Order Reduction for Closed-Loop Systems*, Ph.D. Dissertation, School of Aeronautics and Astronautics, Purdue University, West Lafayette, Indiana, August 1991.

11. Tiffany, S.; and Adams,W.: *Nonlinear Programming Extensions to Rational Function Approximation Methods for Unsteady Aerodynamic Forces Which Allow Variable Selection of Constraints*, NASA TP-2776, May 1988.

12. Tiffany, S.; and Karpel, M.: *Aeroservoelastic Modeling and Applications Using Minimum-State Approximations of the Unsteady Aerodynamics*. NASA TM-101574 and AIAA Paper 89-1188. Presented at the AIAA/ASME/-ASCE/AHS 30th Structures, Structural Dynamics, and Materials (SDM) Conference in Mobile, Alabama, April 1989.

13. Miller, G: *Active Flexible Wing (AFW) Technology*. NA-87-1515L, 1987.

Table 1 1989 and 1991 Simulation Math Models

State Categories	1989	1991
Symmetric flexible mode positions and velocities	16	20
Sym aero lag states associated w/ the flexible modes	8	40
Sym aero lag states associated w/ the control modes	4	16
Symmetric turbulence states	2	2
Anti-sym flexible mode positions and velocities	14	20
Anti-sym aero lag states associated w/ the flexible modes	7	40
Anti-sym aero lag states associated w/ the control modes	4	16
Anti-sym turbulence states	2	2
Linear actuator states, 2 per actuator, 8 actuators	16	16
Coupled linear aeroelastic submodel	**73**	**172**
Non-linear actuator states, 1 per actuator, 8 actuators	8	8
Anti-aliasing filters on 40 channels	40	40
Total states	**121**	**220**
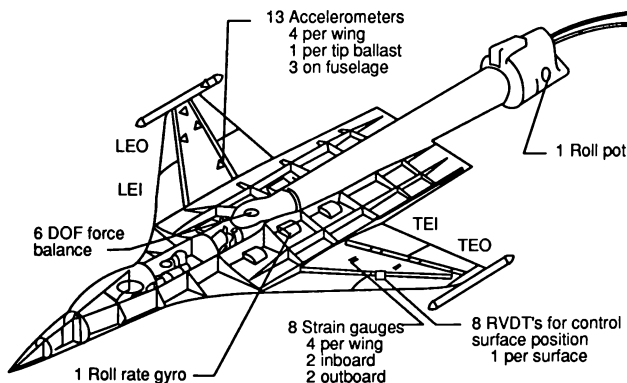
Figure 1.- Instrumentation of the AFW model.
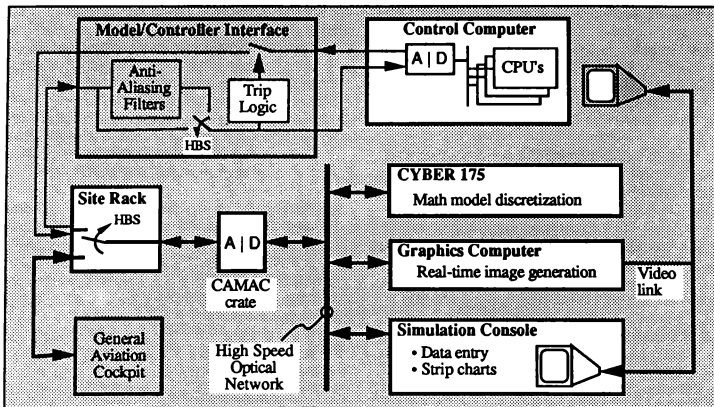
Figure 2.- Schematic of the AFW hot-bench simulation (HBS) laboratory.
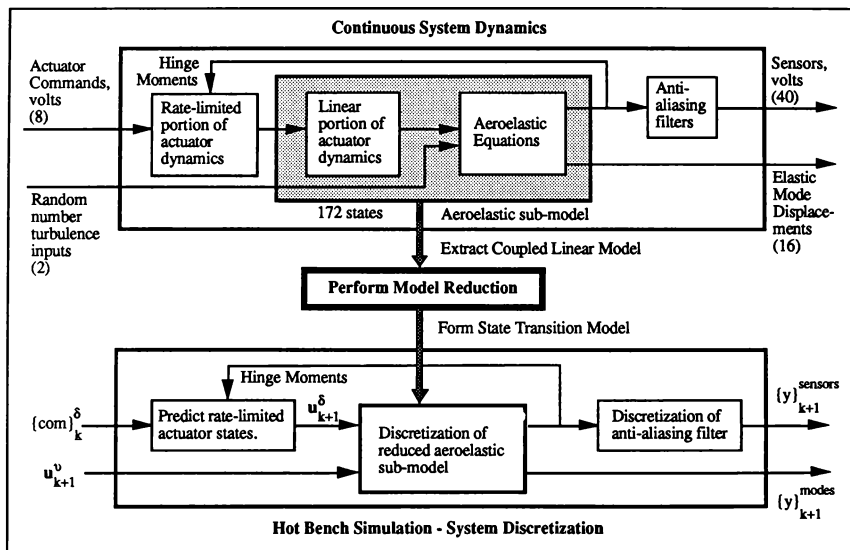
Figure 3.- Data flow from the batch to the hot-bench simulation.
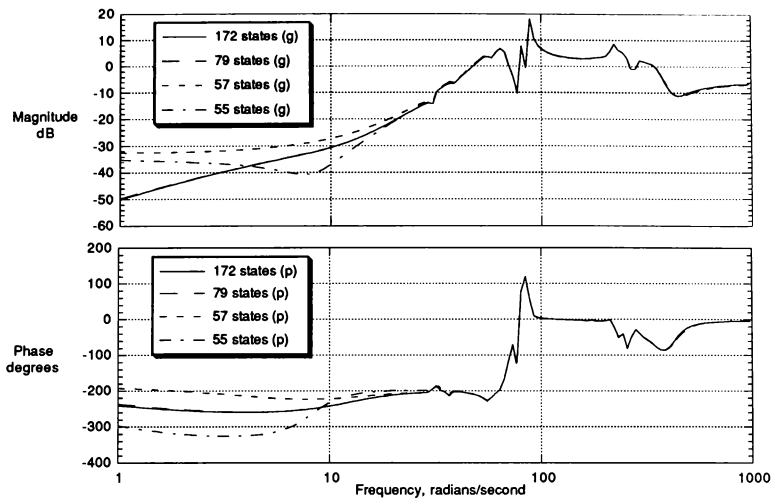
Figure 4. - Frequency response of the right tip accelerometer to commanded right trailing edge outboard control
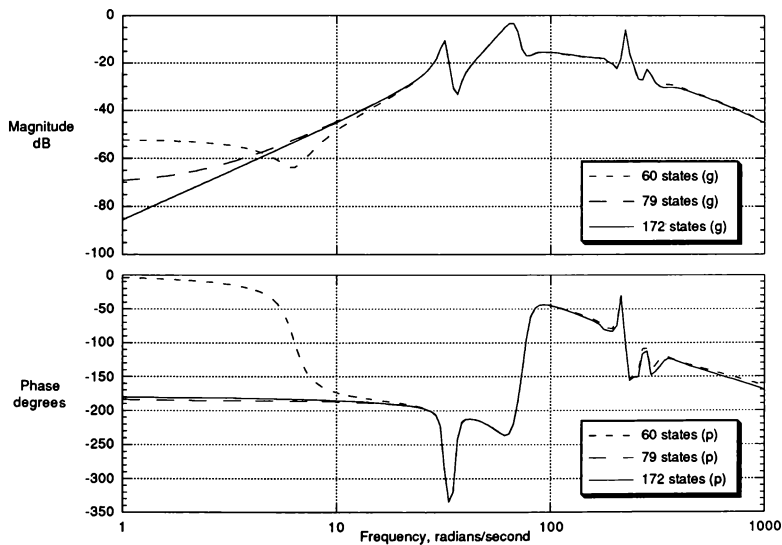
Figure 5. - Frequency response of the right tip accelerometer to symmetric turbulence input

193
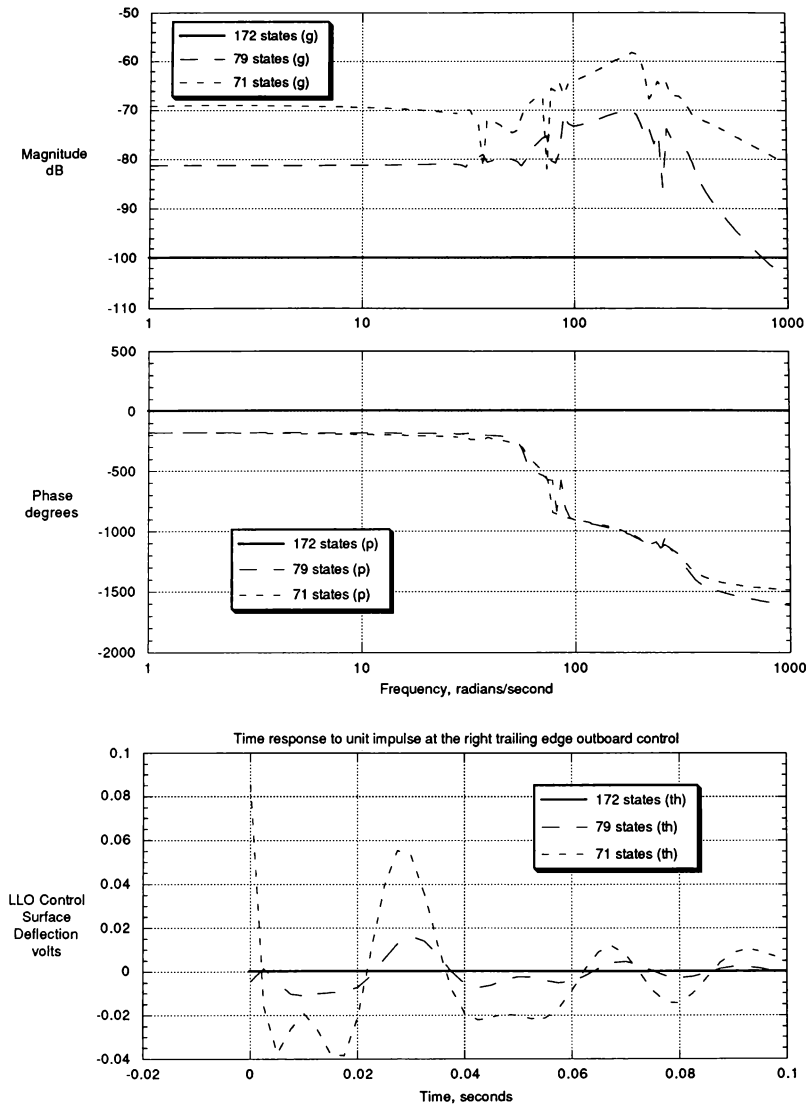
Figure 6. - Magnitude, phase, and time responses of the left leading edge outboard (LLO) control surface to a command at the right trailing edge outboard actuator for various levels of model reduction at dynamic pressure of 300 psf, free-to-roll.
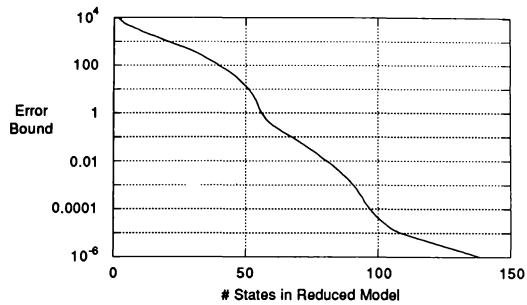
Figure 7. - Error bound vs number of states for the aeroelastic submodel at a dynamic pressure of 300 psf, free-to-roll
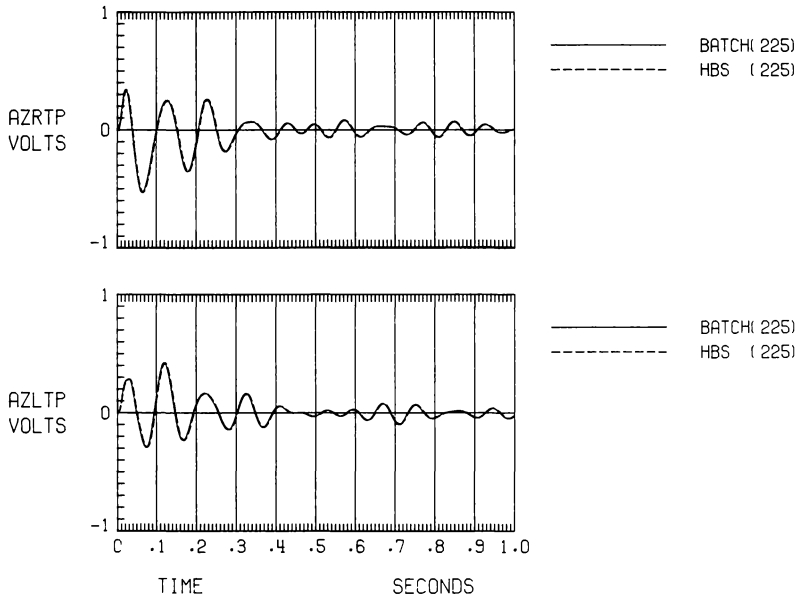
Figure 8. - Comparison between a full order batch simulation and the hot-bench simulation with a 79 state aeroelastic submodel for the right and left tip accelerometers for a 1 volt step command to the right trailing edge outboard control surface at a dynamic pressure of 225 psf, free-to-roll.

SIMULATED LEADSHIP AERODYNAMIC INTERFERENCE AS APPLIED IN THE B-2 AIRCREW TRAINING DEVICE

Jerome M. Weiss*, Section Head (Aero/Propulsion)
CAE-Link Corporation
Binghamton, New York

Abstract

The B-2 Aircrew Training Device (ATD) required high fidelity crew training in aerial refueling and Minimum Interval Takeoff (MITO). This translated into modeling the preceding vehicle's (leadship) vortices and engine exhaust effects on the following vehicle (lagship). Also modeled was the leadship's interaction with lagship's bow wave. These models were developed to yield representative effects independent of the specific vehicles being addressed.

Introduction

The B-2 Aircrew Training Device (ATD) required the modeling of the aerodynamic and propulsion effects of a fixed-wing leadship upon a fixed-wing lagship. These effects are useful for training aerial refueling and/or MITO maneuvers. These effects produce the following crew stimuli:

1. Delta fuel flow

2. Delta trim for 1-G flight

3. Delta aerodynamic controllability

4. Delta rate of climb (descent)

5. Leadship exhaust sensory cues

The training requirements were satisfied by modeling the following effects:

1. Relative position between the leadship and lagship

2. Leadship exhaust effect on the lagship

3. Leadship wingtip and sheet (bound) vortices/downwash effects on the lagship.

4. Leadship interaction with the lagship's bow wave.

Math models were developed for high maintainability and reusability. This was accomplished by separating the vehicle-dependent parameter values from the simulation model and by separating the software

_____

* Senior Member of AIAA

structure from the simulation model. Therefore the models are independent of the fixed-wing vehicles to be simulated. These generic math models were applied to the B-2, KC-135, and KC-10 as leadships and the B-2 as the lagship.

The design goal for these two models was to yield representative values (approximately 75% accurate for any particular application) for this complex phenomenon. The remaining accuracy would be achieved by subjective pilot evaluation since manufacturer-specific data is currently not available.

This paper was written before the subjective pilot evaluation was performed on the B-2 ATD.

The relative position model computes the leadship/lagship relative position in inertial and leadship and/or lagship body reference systems. This model is applied at each lagship reference point. The relative position is used by the vortex/downwash models, leadship exhaust velocity model, and bow wave drag model.

The model of leadship exhaust velocity effects on the lagship is based on past simulation work at Link.[9] This exhaust velocity is a function of distance between the leadship engine(s) and the lagship reference point(s). This model only addresses delta aerodynamic forces and moments which rely directly on the delta in angle of attack (AOA) or its subsequent differential wing lift distribution. No delta in angle of sideslip (AOS) is considered in the exhaust velocity effects model.

The vortex/downwash was modeled using either an approach based on current literature (Literature Approach) or an approach based on a provided database (Database Approach).

The Literature Approach computes the leadship swirl velocities with decay factors for longitudinal relative position and environmental turbulence. This approach does not include any delta AOS effects. It models delta aerodynamic forces and moments which are directly related to the delta in AOA or its subsequent differential wing lift distribution.

This approach was used only for MITO simulation in the B-2 ATD.

The Database Approach is based on a computer-generated flow field of X, Y and Z velocity components within a leadship body reference system. Appropriate decay factors are applied to the velocity components. Then they are used to compute delta AOAs and delta AOSs at each lagship reference point. The leadship aerodynamic interference (AI) effects contain delta aerodynamic forces and moments. These delta aerodynamic forces and moments may contain minimum value components, direct angle of attack/sideslip effects, and aerodynamic rate components. Additional control surface efficiency factors are created for selected aerodynamic coefficients due to control surfaces. This approach was used for aerial refueling in the B-2 ATD.

The leadship bow wave drag model accounts for the bow wave of the lagship interacting with the leadship. This model yields a delta aerodynamic drag force. This drag force typically presents itself in close formation flying such as aerial refueling.

Nomenclature

Symbols

ΔL	delta lift force
ΔD	delta drag force
ΔSF	delta side force
ΔYM	delta yawing moment
ΔRM	delta rolling moment
ΔPM	delta pitching moment
$\Delta \alpha$	delta angle of attack
$\Delta \beta$	delta angle of sideslip
V_p	appropriate local lagship flight path velocity
ΔV_y	decayed delta lateral velocity component
ΔV_z	decayed delta vertical velocity component
S_{rp}/S	representative area ratio for a given reference point
b_{rp}/b	representative span ratio for a given reference point
ΔC_x	delta generalized aerodynamic coefficient
ΔF_x	delta generalized aerodynamic force
ΔM_x	delta generalized aerodynamic moment
ΔXM_F	delta moment due to differential delta aerodynamic forces
$k_{adj_{A-C}}$	adjustment constants A thru C
b	lagship wingspan
S	lagship reference area
q	free-stream dynamic pressure
q/q_∞	dynamic pressure losses ratio

l_y	lagship reference point lateral moment arm
ϵ	pitch and/or lateral control surface efficiency terms
δ	directional control surface efficiency terms
$\Delta x'_B$	longitudinal relative position in the leadship body axis
v_a	lagship lateral velocity in lagship body axis

Subscripts

rp	reference point(s)
v	due to vortices
E	due to exhaust effects
avg	average over all reference points
L	left side reference point(s)
R	right side reference point(s)
α/β	angle of attack (AOA) and/or angle of sideslip (AOS)
max	maximum
stall	value for aerodynamic stall
AI	aerodynamic interference

Description of Math Models

The models described in this paper represent a methodology for computing the leadship effects on the lagship. The models are independent of the fixed-wing vehicles being simulated, and therefore generic. The different models of this simulation and their minimum numbers of lagship reference points are listed below:

1. Relative position between a single leadship reference point and all lagship reference points.

2. Vortex/downwash including both tip and bound leadship vortices.

   a. Literature Approach – one lagship left wing and one lagship right wing reference point.

   b. Database Approach – two lagship left wing and two lagship right wing reference points.

3. Leadship propulsion exhaust effects on the lagship. Exhaust model – one lagship left wing and one lagship right wing reference point.

4. Lagship bow wave drag – one lagship left wing and one lagship right wing reference point.

The goal of these models was to produce representative values (approximately 75% accurate for any particular application) of the delta aerodynamic interference effects of a leadship on the lagship. These effects primarily consist of delta aerodynamic interference forces and moments. Fig. 1 shows the sign convention and lagship reference points used in this paper.
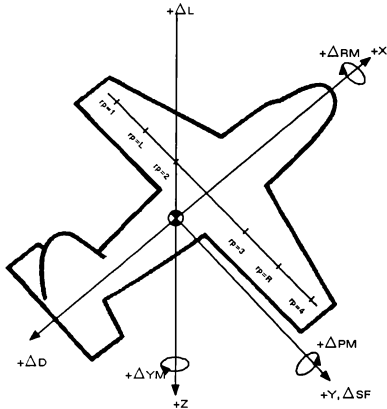
Fig. 1 Sign Convention for Aerodynamic Forces and Moments

The general approach to computing delta aerodynamic interference forces and moments from the various models is the same. This general approach is shown in Fig. 2. The major differences between the Literature and Database Approaches are the techniques for determining the leadship vortex/downwash delta velocities, number of lagship reference points, and the components of the delta forces and moments. Both approaches make a basic assumption that the leadship has a uniform lift distribution. This means that the left and right side vortices are of equal strength. The expected general trends due to the leadship downwash/vortices and bow wave drag on the lagship are described below. This description is for a lagship moving outboard to the right of the leadship centerline but not at vortex core yet and with constant longitudinal and vertical relative position:

1. Lagship rolls back towards the leadship centerline (negative rolling moment)

2. Lagship lift force increases

3. Lagship drag force increases
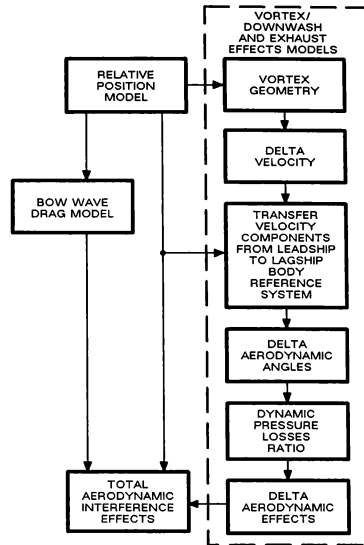
Fig. 2 General Model Flow

4. Lagship pitches nose down (negative pitching moment)

5. Lagship nose yaws to the left (negative yawing moment)

6. Lagship is pushed towards the leadship centerline (negative side force)

The exhaust model's delta velocity is directly from the leadship engine(s). It has a unique feature in that it supplies sensory cues to the crew members, too.

Each model/approach is first discussed in general terms, followed by model/approach specific details as appropriate. The summation of sets of delta forces and moments from these models yields the total delta AI forces and moments applied to the lagship. The total delta aerodynamic forces and moments are summed with the free–stream aerodynamic forces and moments. These summed aerodynamic forces and moments are used to compute the appropriate aircraft equations of motion. The selected control surface efficiency factors are included in the free–stream aerodynamic coefficient simulation expressions.

Relative Position Model

This model will produce the relative position between the two vehicles (leadship and lagship) in the inertial reference frame (I-frame). It will also yield the relative position in either the leadship or lagship body axis reference frame (B-frame). This relative position information is used by all of the other models addressed in this paper.

The I-frame is a rectangular axis system with its origin at the center of the earth. The I-frame X – Y plane forms the equatorial plane and positive Z is through the South Pole. Positive X is directed towards the earth's surface through the Greenwich Meridian (zero longitude) and the positive Y direction completes the right-handed orthogonal triad. This model assumes a spherical nonrotating earth.

This model consists of three major components. The first component contains the direction cosine relationships to transfer the vehicle's center of gravity from the B-frame to the I-frame. The second component computes the vehicle's reference point(s) position vector in the I-frame. The final component computes the relative position between the two vehicles in the I-frame and transfers this relative position back to either the leadship's or the lagship's B-frame.

Note that the first two major components are done for both the leadship and lagship. The relative positions are computed between the single leadship reference point and each lagship reference point. Fig. 3 presents the layout of this model. The reader is reminded that a direction cosine matrix inversion is equal to its transpose.
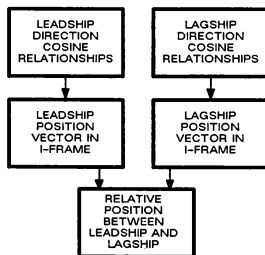
Fig. 3 Relative Position Model[1,16]

When the captured data technique is used (see Literature Approach), an alternative method is used to obtain the leadship's c.g. position in the I-frame.[16] It is assumed that the leadship flies at the captured attitude, altitude, and airspeed for the delta time equal to the difference between the captured data generation time and the current mission time. An estimated leadship c.g. I-frame position is computed based on its captured parameters and the delta time. This allows the vortex/downwash and exhaust effects models to be applied to a mission where the model assumption of similar leadship and lagship flight paths may not be true. The captured data technique may also be used when a significantly large relative position exists.

Relative position of lagship with respect to the leadship in the I-frame is obtained by subtraction of the components between the leadship and lagship. When the I-frame relative position is significantly large, the earth's curvature needs to be addressed. The relative position in either the leadship or lagship B-frame is obtained by multiplying the I-frame relative position with the appropriate leadship or lagship I to B direction cosine relationships. The average relative position between the lagship and the leadship in the leadship B-frame is desired within the vortex/downwash and bow wave drag models. It is also typically needed as an export parameter. Its formulation is the sum of each reference point's relative position divided by the number of reference points.

Leadship Aerodynamic Interference (AI) Models

The delta aerodynamic forces and moments from the vortex/downwash models, exhaust models, and bow wave drag model yield the leadship AI effects on the lagship. The general approach for the vortex/downwash and exhaust models is shown in Fig. 4. Each item in Fig. 4 is discussed for each model/approach in general terms, with model/approach specific details added as required. A brief background including assumptions precedes the detailed model discussions. The bow wave drag model is discussed last since it doesn't fit into the general approach of the vortex/downwash and exhaust models.

Literature Approach – Background – This approach is the result of an extensive data search through the Link Information Center.[2-8] At a minimum the lagship should have two reference points (one on the left wing and one on the right wing) to enable the model to account for differential lift and drag effects. This approach did not address AOS. Based on this, the side force effects are considered negligible. This approach is most useful when no flow field data is provided by either the aircraft (lagship) manufacturer or the customer. The B-2 ATD uses this approach for MITO only.

```
┌─────────────────────────────────────────────┐
│ VORTEX GEOMETRY                               │
│  • VORTEX LONGITUDINAL ROLLUP DISTANCE        │
│  • VORTEX CORE LATERAL POSITION               │
│  • VORTEX CORE VERTICAL POSITION              │
└─────────────────────────────────────────────┘
                      ↓
┌─────────────────────────────────────────────┐
│ DELTA VELOCITY                                │
│  • SWIRL VELOCITY                             │
│  • DELTA VELOCITY RATIO                        │
│  • EXHAUST VELOCITY                           │
│  • VORTEX DELAY FUNCTIONS                      │
└─────────────────────────────────────────────┘
                      ↓
┌─────────────────────────────────────────────┐
│ TRANSFER VELOCITY COMPONENTS FROM             │
│ LEADSHIP TO LAGSHIP BODY REFERENCE SYSTEM     │
└─────────────────────────────────────────────┘
                      ↓
┌─────────────────────────────────────────────┐
│ DELTA AERODYNAMIC ANGLES                      │
└─────────────────────────────────────────────┘
                      ↓
┌─────────────────────────────────────────────┐
│ DYNAMIC PRESSURE LOSSES RATIO                 │
└─────────────────────────────────────────────┘
                      ↓
┌─────────────────────────────────────────────┐
│ DELTA AERODYNAMICS EFFECTS                    │
│  • AREA/SPAN RATIOS                            │
│  • DELTA AOA/AOS COMPONENTS                     │
│  • DELTA ANGULAR RATE COMPONENTS              │
│  • EFFICIENCY GAINS ON THE AERO COEFFICIENTS  │
│    DUE TO PRIMARY CONTROL SURFACES            │
│  • MINIMUM VALUE COMPONENTS                    │
└─────────────────────────────────────────────┘
```

Fig. 4 General Flow of Vortex/Downwash and Exhaust Effects Models

The development in the reference materials[2-8] assumes that the leadship vortices extend approximately straight back from the leadship. If the leadship turns, all of the previously generated vortices turn too. Therefore the leadship and lagship must be at approximately the same flight path throughout a maneuver. This may be true for aerial refueling but it is generally not true for the MITO case. In a MITO the lagship may start out directly behind the leadship but then have the leadship, and its previously generated vortices, turn away. The basic equations in the Literature Approach can handle this, but the geometry/tracking task becomes complex. To handle the MITO case a method of recording past leadship positions and its appropriate parameters was developed. This method is called a captured data file technique. The B-2 ATD used leadship information captured in a file for the last three minutes of the simulated mission. The captured data is kept in a circular buffer queue-type file receiving data at constant time increments of one per second. The file is examined to determine the closest recorded data group in front of the current lagship position. The captured data is used by the relative position model and the Literature Approach.[16]

Potential Flow Database Approach – Background
– This approach is based on the results of a supplied

vortex flow field database.[10] A KC-135A leadship vortex flowfield was computed using the Configuration Data Management System (CDMS) and the QUADPAN aerodynamic code. QUADPAN is an analytical panel method program used to solve inviscid, irrotational, linearized subsonic flow problems.[10] These constraints do not allow for viscous decay of the wingtip vortices, vertical movement of the vortex core, or separation of flow behind the fuselage.[10] This database extends slightly beyond the initial rollup stage. The B-2 ATD used only the required (lateral and vertical) delta velocity ratios for the real-time simulation.

The real-time application of the supplied database was questionable due to the large number of linear interpolations required for each delta velocity ratio at each lagship wing reference point. Data reduction led to a method of breaking the single 5-variable function into two functions of 3 and 2 variables. It also reduced the grid size from 33,000 points to approximately 3,100 points.[16]

A method of transforming the KC-135 database into a KC-10 database was obtained by making geometric and vortex strength adjustments.[16] The Database Approach was used only for aerial refueling training on the B-2 ATD.

Exhaust Effect Model – Background

The exhaust of a leadship may present localized high energy air flow over portions of the lagship, particularly the wing. These localized high energy air pockets could result in a change in the lagship attitude and position. Obviously these exhaust effects are a function of leadship engine geometry, leadship/lagship relative position and attitude and leadship power setting (exhaust velocity profile). This model does not include AOS effects. This model's level of completeness of delta forces and moments and the number of reference points are the same as the Literature Approach.

This model also includes sensory cue effects to the lagship crew when the exhaust effects exist. These cue effects will take the form of buffet motion and aural cues. The buffet would be short wave length and of variable amplitude. The amplitude can vary from moderate strength at maximum exhaust velocity to zero at some low exhaust velocity such as 20 fps. This was implemented through the B-2 ATD motion system. Aural cue gain will also vary with these same parameters too. A generic gain was developed for each side (rp = L,R) of the lagship as a ratio of actual leadship exhaust velocity to maximum possible exhaust velocity.[12] This will be used as inputs to the aural cue and motion systems. The aural cue gain in-

cludes an extra term to allow aural cues to occur beyond the low exhaust velocity value. The initial value of these gains will be zero because they are presently not based on aircraft data. They will be adjusted as necessary during subjective crew evaluation. The left and right gain was channeled to the appropriate side of the ATD.

Bow Wave Drag Model

The bow wave drag is applied only when the lagship is very close to the leadship (e.g., aerial refueling – at or near the pre-contact position or boom envelope). A generic approach is used to determine the delta drag between the free- stream and close proximity drag forces. This delta drag then has the exhaust/vortex effects removed. The result is the bow wave drag. The close proximity drag force can be assumed to be equal to the thrust force obtained from the fuel flow published for close proximity operations (e.g., Performance Appendix to Flight Manual). In the B-2 ATD application the bow wave drag force was supplied by the aircraft manufacturer.[17] The supplied expression is believed to be empirically derived. The expression is a function of the leadship and lagship weights, dynamic pressure and leadship wingspan. The supplied bow wave drag force is multiplied by a fade-in/out function to allow a smooth application of this force. Historically this effect is noticed during refueling by lagship crews. It appears to start when the lagship is approximately 50 feet from the boom envelope aft limit. This effect ramps up linearly as the lagship approaches the leadship until just prior to the boom envelope (aerial refueling), then drops sharply to a constant value equal to the supplied bow wave drag value. This drop was arbitrarily selected to be 30% of the peak value. Pilots fly through this peak with a short and quick throttle movement. The bow wave drag expression includes an adjustment constant to aid in the subjective pilot evaluation.

Detailed Model Components

Vortex Geometry

The first item in Fig. 4 addresses the required knowledge of the vortex position relative to the lagship reference points. The required vortex geometry consists of the following:

1. Vortex Longitudinal Rollup Distance – The methodology used to compute this distance assumes an elliptic leadship lift distribution. This distance defines the length for the wingtip vortex to be fully developed and is based on Eq. 27 of Ref. 13. The final longitudinal dis-

tance for rollup is adjusted so that it is measured from the leadship reference point.

2. Lateral Distance for Vortex Rollup – Several references give equations for the lateral vortex separation when the vortices are rolledup. This application used Eq. 2 of Ref. 2, which yielded a lateral separation of 78.5% of the leadship wingspan. It is assumed that the vortex cores move linearly from the leadship wingtips to the lateral separation distance for rolled-up vortices.

   To take the effects of crosswinds during a MITO into account, the crosswind components must be transformed from the leadship E-frame to the lagship B-frame. It is assumed that the horizontal steady wind components are the same for both the leadship and the lagship. This effect is typically used for MITO (low speed) only and not for aerial refueling (high speed). The equations used to predict the lateral position due to crosswinds are based on Refs. 2 and 16.

3. Vertical Position Information – The methodology to compute this distance is based on the vortex rate of descent found in Refs. 2 and 5. A special consideration is included for MITO where the ground reflective plane influences the vertical height of the vortex cores. Therefore the vortex cores were not allowed to drop below a height less than one-half the leadship wingspan. The geometric altitude of the vortex core was computed as a function of the leadship altitude, geometry, vortex core rate of descent and delta time between the leadship and lagship.[16]

Delta Velocity

The leadship aerodynamic interference effects due to the vortex/downwash models and exhaust effects model depends on the computation of a delta velocity. The delta velocity for the vortex/downwash models is required to be multiplied by a total decay factor. This is true since the delta velocity equations do not apply in all stages of Fig. 5. The decay factor development is presented in generic form supplemented by application-specific information. Exhaust effects do not extend as far as the vortex/downwash effects and the velocity profiles contain natural decay of the velocity. Therefore the total decay factor is not included in the exhaust effects model. The various models require different approaches to the delta velocity computations.
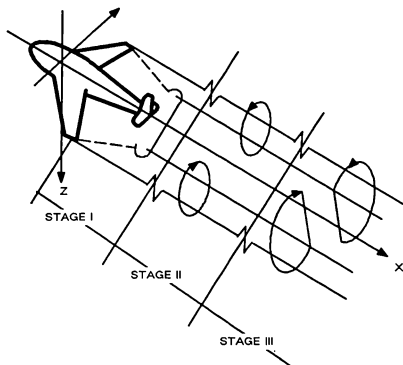
Fig. 5 Development Stages of Trailing Vortices

1. Literature Approach to Swirl Velocity – Two different equations are used to compute the swirl velocity: viscous flow solution and potential flow (Biot– Savart) solution. The determination of which equation to be used is a function of the vortex core radius and the radial distance of a reference point to the vortex core.[2,6,7] This provides a natural way to account for radial (y and z) decay of the vortices. The vortex core radius equation was studied from various references.[2,4,5,6] A significant amount of scatter existed for this parameter when the various equations were exercised for a given point in the sky. This led to the use of an average of these empirically derived equations. The Ref. 2 simplified expression of the viscous solution is used when a reference point is within the vortex core diameter. The Ref. 2 potential flow solution is used when the reference point is outside of the vortex diameter. The swirl velocity used the leadship wingtip circulation strength at altitude and the total decay factor.[16] A discussion of these terms is presented below. The swirl velocity must be computed for each of the leadship wingtip vortices at each reference point on the lagship. The sum of these ·swirl velocities at each lagship reference point becomes the total swirl velocity at that reference point. The swirl velocity is computed in the leadship body axis system.

The leadship circulation strength assumes a uniform elliptic lift distribution. It also assumes that only one pair of tip vortices persists behind the leadship. The circulation for both leadship wingtips is necessary for a complete representation of this effect. In this model the left and right wingtip strength are equal since the circulation is uniform. The circulation strength is computed from the equations for the Ref. 2 clean configuration. Using Ref. 2 and Ref. 5, a leadship wingtip circulation at sea level could be modeled as a function of leadship gross weight, normal acceleration, airspeed, and reference weight. Additionally, a methodology is presented in Ref. 2 to take the effect of altitude into account. Using this approach, the leadship circulation at any altitude was computed.[16]

2. Delta Velocity Ratios – The basic delta velocity ratios were obtained from the supplied flowfield analysis[10] after it was put into a real–time format. The basic delta lateral and vertical velocity ratios were then multiplied by the total decay factor to obtain the decayed delta velocity ratios. These computations were repeated for each of the four lagship wing reference points. The total decay factor is derived below.

3. Vortex Decay Functions – Two different types of decay are addressed, decay due to distance and decay due to environmental effects. The product of the two decay factors yields the total decay factor. The total decay factor is used to compute the actual swirl velocity or delta velocity ratio. For this calculation, the left and right vortices are handled in the same manner. No decay function exists for the exhaust model.

   a. Vortex Decay Function – Longitudinal Decay Factor – The average longitudinal relative position distance is the only distance considered.

   A three–stage approach is assumed in this application.[4] The three stages are shown in Fig. 5 and are identified as follows:

   Stage I – Initial rollup (bound vortex rolls into wingtip vortex, lateral movement stabilized and wingtip circulation fully developed)

Stage II – Constant circulation for a particular airspeed and altitude (vortex radius increases from initial rollup value until the left and rightwingtip vortices make contact).

Stage III – Vortex breakup (left and right wingtip vortices radii overlap).

The longitudinal decay factor in Stage I only applies to the Literature Approach because the provided database used in the Database Approach covers the distance from the leadship to vortex rollup. The longitudinal decay factors for Stage II and III and the environmental decay factor apply to both vortex approaches.

In Stage I the swirl velocity is linearly increased from half strength to full strength of the swirl velocity.[4]

Stage II region covers from the end of Stage I to the start of vortex breakup. This distance corresponds to the time duration of Stage II given in Refs. 2,5,6,15. This time has a maximum value of 120 seconds. This time value is a function of the leadship and the value selected here is the largest time value given. Within this region the swirl velocity/delta velocity ratio is a constant for a given airspeed and position.[2,4] Therefore the Stage II longitudinal decay factor is unity.

Stage III is the region where the left and right vortex radii overlap. The duration time of 30 seconds is used to compute the length of Stage III.[5] During Stage III the swirl velocity/delta velocity ratio is linearly decreased to zero. This decay is probability logarithmic in reality, but in this simulation it is assumed to be linear.[4]

At typical refueling speeds the relative position for Stages II and/or III can be significant. Therefore the earth's curvature must be accounted for in the relative position model. Another approach would be to apply a captured data technique to Stages II and/or III.

b. Vortex Decay Function –Environmental Decay Factor – The environmental turbu-lence adds energy to the air, therefore increasing the rate of decay from calm air. Data for this change in the decay rate is obtained from Ref. 6. This reference presents time to vortex bursting as a function of environmental turbulent dissipation rate during takeoff. The data was made into a percent factor to be multiplied with the current stage circulation decay term. In an effort to expand the data beyond the takeoff maneuver (low airspeed and altitude) the percent factor was made a function of leadship altitude. The higher the altitude is, the less turbulence energy exists for decay. An arbitrarily selected decrease of 10% is placed on this data for high–altitude operations. The percent factor ranges from 1.0 for negligible turbulence to approximately 0.035 for moderate turbulence.[6,16] No correction was made to this data to compensate for flap/no-flap configurations since this model only assumes a single pair of wingtip/bound vortices. The definition of high and low altitude is selected to be a ballpark value of the thermo–mixing layer. A value of 4,000 ft AGL was selected. The environmental decay factor is ramped in over the last half of the Stage II region and is applied in full strength throughout Stage III.[2,6] The total vortex decay factor is the product of the longitudinal and environmental factors.

4. Leadship Exhaust Velocity – Engine exhaust velocity is normally a function of power setting, outside air temperature, and distance from the engine exit plane. Remembering the design goals, this model makes the following assumptions:

a. Leadship power setting remains constant during both the aerial refueling and MITO maneuvers.

b. Leadship typical turbojet exhaust velocity profiles are independent of the particular vehicle. The same velocity profile is used for all leadships. This assumes that the delta in thrust classes is primarily accounted for in delta mass flows, not delta velocity profiles.

c. Takeoff power setting is used for a typical MITO velocity profile.

d. Seventy percent of the takeoff power setting is used for a typical aerial refueling velocity profile.

e. The typical velocity profiles assume symmetry in the Y and Z velocity profile directions.

Logic examines the relative position to determine which leadship engine(s) is close enough to have an effect on a given lagship wing reference point. Since the different power setting exhaust velocity data is mission-dependent, the lagship height above ground (AGL) is used to determine which data is appropriate. An arbitrary altitude of 3,000 ft AGL is used as the decision altitude. If no leadship engine is determined to have an effect on the lagship, then the exhaust forces, moments, and cues are set equal to zero.

The total exhaust velocity vector is broken into longitudinal, lateral, and vertical components. The method used to compute the velocity components depends on the component's percentage of the total distance from the exhaust plane to the lagship reference point being considered.[9] These percentages are multiplied by the total exhaust velocity vector, yielding the desired exhaust velocity components. This is done for all lagship wing reference points.

Transfer Velocity Components from Leadship to Lagship Body Reference System

The third step of Fig. 4 transfers the above delta velocity components from the leadship B-frame to the lagship B-frame. If these velocity components are to be used to compute lagship delta AOA and AOS, they must be in the lagship B-frame. This transformation is accomplished in two major steps:

1. Transforming the velocity components from the leadship B-frame to the I-frame using the transpose of the leadship I to B frame direction cosines matrix.

2. Transforming the velocity components from the I-frame to the lagship B-frame using the lagship I-to-B frame direction cosine matrix.

The following model-specific comments apply:

The Literature Approach assumes that its delta velocity is totally a vertical velocity[2] in the leadship B-frame.

In the Database Approach the delta longitudinal velocity ratio was found to be approximately zero.

No lagship B-frame lateral exhaust velocity components were considered because AOS effects are not computed in the exhaust effects model.

Delta Aerodynamic Angles

The technique for computing the change of lagship AOS and AOA depends on the appropriate decayed delta velocity component.[2,6] This technique is applied to each lagship wing reference point for each appropriate delta velocity source (e.g., Exhaust Model – n engines, Literature Approach – both leadship wingtips). The Database Approach is the only approach that produces a delta AOS and delta AOA. The Literature Approach and Exhaust Models produce only delta AOA. Note that only one set of vortex delta aerodynamic angles are computed and, if appropriate, a separate delta AOA due to exhaust is computed. Subscripts v and E are used to denote the delta angle due to the leadship vortices and/or exhaust.

The change of lagship AOS uses the lateral decayed delta velocity. The lagship lateral velocity and the lateral decayed velocity are applied to the standard AOS equation to yield a combined AOS. The delta AOS at a reference point is the difference between the combined AOS and the lagship AOS. Note that the combined AOS is used to obtain the lagship total velocity projection onto the X-Z plane.

The following are the general equations for computing the delta AOA, AOS, and combined AOS at each lagship wing reference point:

Eq. 1   Combined AOS and Delta AOS

$$\beta_{rpv} = \sin^{-1}\left(\frac{v_a + \Delta V_{yrp}}{V_p}\right) \; ; \; \Delta\beta_{rpv} = \beta_{rpv} - \beta$$

where: ΔV_{yrp} = decayed lateral delta velocity computed by Database Approach

Eq. 2   Delta AOA

$$\Delta\alpha_{rpv,E} = \sin^{-1}\left(\frac{\Delta V_{zrp}}{V_p \cdot \cos\beta_{rpv}}\right)$$

where: $\Delta V_{r_{rp}}$ = decayed vertical delta velocity due to all leadship vortex or exhaust sources at a reference point.

Eq. 3 The average AOA and AOS

$$\Delta a_{avg_{v,E}} = \sum \Delta a_{rp_{v,E}}/\text{number of reference points}$$

$$\Delta \beta_{avg_v} = \sum \Delta \beta_{rp_v}/\text{number of reference points}$$

The Database Approach requires additional expressions for average left and right side delta AOAs and AOSs.

Dynamic Pressure Losses Ratio

When a lagship is operating near a leadship, use of the free-stream dynamic pressure to compute aerodynamic relationships is not accurate. A reduction in the free-stream dynamic pressure is caused by the loss of flow energy in the form of friction and separation drag of the leadship.[11] A dynamic pressure losses ratio is computed to account for this effect. This ratio is a function of leadship friction, geometry, and relative position. The following assumptions are used by Refs. 11 and 16:

1. Operations occur only in the linear range of angle of attack.

2. The spanwise variation of this ratio is not significant compared to its longitudinal variation.

3. The average relative position can be used instead of each reference point relative position.

A ratio of dynamic pressure is computed according to para. 4.4.1 of Ref. 11, and is multiplied by the free-stream dynamic pressure to obtain the correct dynamic pressure relationship.

This paper also considers the change in dynamic pressure due to the leadship exhaust velocity adding energy into the free-stream. This is handled by determining a lagship delta dynamic pressure based on the delta exhaust velocity at the lagship wing reference point(s) and adding this to the lagship free-stream dynamic pressure. The exhaust dynamic pressure is computed using the standard dynamic pressure equation with the total delta exhaust velocity.

Delta Aerodynamic Effects

The last step in Fig. 4 computes the delta aerodynamic forces and moments as well as control surface efficiency factors. These computations, along with the bow wave drag, yield the total delta AI of a leadship on a lagship. The generic expressions used to compute the total delta AI effects from the vortex/downwash and/or exhaust effect models contain some or all of the following components:

1. Area/span ratios

2. Delta AOA/AOS components (Eq. 4 – Eq. 6)

3. Delta angular rate components

4. Efficiency gains on the aerodynamic coefficient due to the primary control surfaces (Eqs. 7 and 8)

5. Minimum value components

The various components for either vortex/downwash approach and exhaust effects model are summarized in Table 1. A discussion about each type of component is presented below.[16] The model-unique aspects of any component's general equations are limited to the appropriate area/span ratios.

1. Area/Span Ratios – To appropriately apply dependent coefficient AOA information (e.g., $C_{L\alpha}$) at each reference point, information about the lagship lift distribution is required. This information is used to determine what percentage of the supplied total vehicle coefficient component should be allotted to each reference point. This type of information is not typically available. Therefore the area ratios are used to compute the lift and drag forces. A similar approach is taken for the supplied AOS-dependent coefficient terms (e.g., C_{l_β}, C_{n_β}).

The area (S_{rp}/S) or span (b_{rp}/b) ratio per reference point is the ratio of the area or span being represented by that reference point divided by the total aircraft area or span. For example, if a reference point represents the entire left wing, then its area ratio has a value of 0.50.

2. Delta AOA/AOS Components – The delta aerodynamic coefficients due to a delta AOA or AOS per reference point are computed by a method which multiplies the delta AOA/AOS by the appropriate coefficient slope and an adjustment constant. The corresponding force or moment will apply the appropriate area ratio for force computations and span ratio for the moment computations. The generic equation form is given below:

Eq. 4 Components due to aerodynamic angles – delta coefficients

$$\Delta C_{x_{rp}} = \Delta(\alpha \text{ or } \beta)_{rpv} \cdot C_{x_{\alpha \text{ or } \beta}} \cdot k_{adj_A}$$

where: $k_{adj_A} = 1.0$

Eq. 5 Components due to aerodynamic angles – delta forces and moments

$$\Delta F_{X_{(\alpha/\beta)rp}} = \Delta C_{x_{rp}} \cdot S \cdot q_{E/\infty} \cdot (q/q_\infty) \\ \cdot (S_{rp}/S)$$

$$\Delta M x_{(\alpha/\beta)rp} = \Delta C_{x_{rp}} \cdot S \cdot q_{E/\infty} \cdot (q/q_\infty) \\ \cdot (b_{rp}/b) \cdot b$$

The delta aerodynamic pitching moment and side force computations use the above technique but without the area or span ratio because they use an average delta AOA/AOS respectively and due to symmetry. The delta drag coefficient per reference point is computed using the delta lift coefficient and the traditional induced drag equation at each reference point. This delta drag coefficient is then used in Eq. 5 to yield the delta drag force per reference point.

The total delta yawing and rolling moment computations may include the effects of delta differential drag and lift force respectively. This generic equation is shown below:

Eq. 6 Delta yawing and rolling moment components – delta differential forces

$$\Delta X M_{F_v} = \left(\Delta F_{x_L} \cdot l_{y_L} \right) \pm \left(\Delta F_{x_R} \cdot l_{y_R} \right)$$

3. Delta Angular Rate Components – Previous subjective pilot evaluation indicated the need to adjust the aerodynamic damping characteristics of the lagship. Therefore a delta roll, pitch, and yaw rate is needed for each reference point. Because these terms are based on previous subjective pilot evaluation only, an arbitrary adjustment constant is included in each of the delta rate term developments. The computed delta rate components are computed in traditional aerodynamic coefficient equation formats. This uses the actual manufacturer's partial aerodynamic coeffi-

cient, the delta rate, and geometry terms to make it non-dimensional. The arbitrary adjustment constant values were found during initial testing. Their values yield results that are consistent with the general model design goals.[16]

Ref. 14 was used to generate the delta roll rate term. This term is a function of the difference in the left and right side delta AOAs. The delta AOA difference between the left and right side is used to represent the differential delta lagship lift distribution. The delta pitch rate is computed in a similar manner, except it uses the average delta AOA. Delta yaw rate is computed by using the difference in the left and right lagship delta AOA and a delta drag to delta lift ratio. This assumes that the delta yaw rate is directly related to the differential delta drag force.

4. Control Surface Efficiency Gains – Previous pilot evaluation of this phenomenon tells us that the controllability of the lagship appears to change inside of this vortex flow field. To account for this, an efficiency term is applied to certain regular aerodynamic coefficient components. Since this is an area where data is usually not available, its form is assumed but its initial value is unity, so as not to affect the regular aerodynamic coefficient components. The form of the pitch and lateral control surface efficiency factor is as follows:

Eq. 7 Pitch/lateral control surface efficiency term

$$\epsilon_{L,R} = 1 - \left| \frac{\Delta \alpha_{L,R_v}}{\alpha_{stall}} \right| \cdot k_{adj_B}$$

where: $k_{adj_B} = 0.0$

The form of the directional aerodynamic coefficient component is shown below. Note that this left/right (L,R) notation is not significant for a conventional rudder configuration; it is more for a spoiler (airbrake) system used for directional control.

Eq. 8 Directional control surface efficiency term

$$\delta_{L,R} = 1 - \left| \frac{\Delta \beta_{L,R_v}}{\beta_{max}} \right| \cdot k_{adj_C}$$

where: $k_{adj_C} = 0.0$

Table 1   Delta Forces and Moments Components

Aerodynamic Degree Of Freedom	Minimum Value	AOA	AOS	Δ_p	Δ_q	Δ_r	Effective Gains On Control Surfaces			Differential Force		Induced
							LONG	LAT	DIR	LIFT	DRAG	DRAG
Lift Force	DB	DB Lit Exh										
Drag Force	DB							DB	DB			DB Exh Lit
Side Force	DB		DB									
Pitching Moment	DB	DB Exh Lit			DB		DB					
Rolling Moment				DB	DB				DB	Exh Lit DB		
Yawing Moment				DB		DB			DB		Exh Lit DB	

Model Notation: Lit – Literature Approach (MITO)
DB – Database Approach (Aerial Refueling)
Exh – Exhaust Effects (Both Maneuvers)

5. <u>Minimum Value Components</u> – The last type of component used, where appropriate, is a delta minimum value component (e.g., C_{Lo}). This consists of an arbitrarily selected linear washout function multiplied by the appropriate manufacturer's supplied minimum coefficient value and an adjustment constant. The adjustment constant is included because this component is based on previous subjective pilot evaluations. The arbitrary adjustment constant values were found during initial testing. Their values yield results that are consistent with the general model design goals.

Table 1 contains all the possible components for any aerodynamic degree of freedom. For any given model (e.g., exhaust velocity) and degree of freedom (e.g., drag) the reference point total delta AI effect contains the components identified on that appropriate row. The total delta forces and moments are the algebraic sum of the various components acting at the lagship reference points.

Results

This paper has described generic engineering math models that compute the leadship effects on a lagship. In this application the leadship could be a KC–135, KC–10, or B–2 depending on the maneuver being trained. The lagship is the B–2. These models were made generic by separating vehicle specific information from the simulation algorithms. The general flow of the simulation is shown in Fig. 2. General flow diagrams for certain models are presented in Figs. 3

and 4. The following generic math models were described:

1. Relative position between a lagship and a leadship.

2. Leadship vortex/downwash effects on a lagship.

    a. Literature Approach

    b. Database Approach

3. Leadship exhaust velocity effects on a lagship.

4. Lagship bow wave drag effects

Testing prior to subjective pilot evaluation resulted in demonstrating that the expected general trends described below were obtained. Therefore the general design goal, approximately 75% accuracy for a given application, was accomplished. The effects of the various longitudinal and environmental decay factors were also exercised, with good results. Following is a summary of the general expected trends when a lagship is outboard (to the right) of a leadship centerline but not at the vortex core:

1. Lagship rolls back towards the leadship centerline.

2. Lagship lift force increases.

3. Lagship drag force increases.

4. Lagship pitches nose down.

Table 2 Typical Exported Parameters

Math Model Symbol	Description
ΔL_{AI}	Total delta lift force due to aerodynamic interference
ΔD_{AI}	Total delta drag force due to aerodynamic interference
ΔPM_{AI}	Total delta pitching moment due to aerodynamic interference
ΔRM_{AI}	Total delta rolling moment due to aerodynamic interference
ΔYM_{AI}	Total delta yawing moment due to aerodynamic interference
ΔSF_{AI}	Total delta side force due to aerodynamic interference
$\Delta X'_B$	Average relative position vector of leadship with respect to leadship wingtip in the leadship B–frame
Control Surface Efficiency Factors	Factors applied to the primary pitch, roll, and yaw control surfaces. This effect is due to the vortex of the leadship. It is included in the drag and all three moments for the Database Model.
Exhaust Effects Cue Flags	Aural cue and motion cue effects flags
$\Delta \alpha_{AI}$	Mean delta AOA due to leadship AI on the lagship in the lagship B–frame (based on vortex and/or exhaust effects)
$\Delta \beta_{AI}$	Mean delta AOS due to leadship AI on the lagship in the lagship B–frame

5. Lagship nose yaws to the left.

6. Lagship is pushed towards the leadship centerline.

7. Leadship exhaust produces sensory cues on logship.

The delta total AI effects computed by these models are listed in Table 2. These are typical exports from these models to be included in the summation of the vehicle external forces and moments.

References

1. "B-52G PPU Relative Position" C. Rockwell, Link SDMR 111033, Sept. 1984.

2. "A Method for Assessing the Impact of Wake Vortices on USAF Operations", G. Kurylowich, AFFDL-TR-79-3060, A072967, July 1979.

3. "Measurements of Boeing 747, Lockheed C-5A and Other Aircraft Vortex Wake Characteristics by Tower Fly-By Techniques", Leo Garodz, Proceedings of Symposium on Aircraft Wake Turbulence, A72-01643, Sept. 1970.

4. "Decay of a Vortex Pair Behind an Aircraft", J. Nielsen & R. Schwind, Proceedings of Symposium on Aircraft Wake Turbulence, A72-01643, Sept. 1970.

5. "Results of the Boeing Company Wake Turbulence Test Program", P. Condit & P. Tracy, Proceedings of Symposium on Aircraft Wake Turbulence, A72-01643, Sept. 1970.

6. "Measurement of the Trailing Vortex Systems of Large Transport Aircraft Using Tower Fly-By and Flow Visualization", L. Garodz, D. Lawrence, N. Miller, FAA-RD-75-127, AD A021305, Jan. 1976.

7. "Vortex Wake of Conventional Aircraft", Coleman DuP. Donaldson, AD A011605, AGARD-AG-204, May 1975.

8. "Vortex Wake Development and Aircraft Dynamics", J. Hackett & J. Theisen, Proceedings of Symposium on Aircraft Wake Turbulence, A72-01643, Sept. 1970.

9. "ASUPT – 74 Vol. II Aerodynamic and Flight Performance", A. Pratt, Singer-Link, May 1974.

10. "Flowfield Analysis Study", USAF/ASD/ENF-TA, August 1988.

11. "USAF Stability and Control – Datcom", McDonnell Douglas Corp., Douglas Aircraft Division, Sept. 1970.

12. "The Handling and Performance Trials Needed To Clear An Aircraft To Act As a Receiver During Air to Air Refueling", AGARD Conference Proceedings No. 373, Flight Test Techniques, AGARD-CP-373, AD A 147 625, 1984.

13. "The Rolling Up of the Trailing Vortex Sheet and Its Effects on the Downwash Behind Wings", Spreiter and Sacks, Journal of Aeronautical Sciences, Jan. 1951.

14. "Airplane Flight Dynamics and Automatic Flight Controls", Jan Roskam, Roskam Aviation and Engineering Corp., 1982.

15. "Flight-Test Evaluation of the Wing Vortex Wake Generated by Large Jet-Transport Aircraft", W. H. Andrews, G. H. Robinson, G. E. Krier and F. J. Drinkwater III, N70-40912, 1970.

16. "Othership System, Software Detailed Design Document", J. M. Weiss, Link SDDD BD100-0101-02, April 1990.

17. "Tanker Interference with Receiver Aircraft", 190-89-091, Northrop Corp., August 15, 1989.

PROPULSION MODELING TECHNIQUES AND APPLICATIONS FOR THE NASA DRYDEN X-30 REAL-TIME SIMULATOR

John W. Hicks*
NASA Dryden Flight Research Facility
Edwards, California

Abstract

The current National Aero-Space Plane (NASP) Program Phase 2 design and development process includes developing flight-envelope expansion techniques, with the definition of range requirements, flight maneuver techniques, and abort scenarios. This is being accomplished with the NASA Dryden NASP Engineering Simulator (NES) by the Edwards AFB government flight test team. The team is made up of NASA Dryden Flight Research Facility and Air Force Flight Test Center personnel. Because the current contractor simulation models focus on the single-stage-to-orbit (SSTO) mission, the Edwards team has developed unique real-time propulsion modeling techniques to perform the necessary simulation flight studies. Flexible, multifaceted propulsion models were developed to allow for flight operational assessment of design features or to perform design trade studies. These models account for expanded operating conditions and internal propulsion performance characteristics, and include advanced engine throttling concepts. By using these techniques, excellent results were obtained in producing high-fidelity, advanced ramjet–scramjet propulsion models that were structurally simple and computationally fast for real-time application. Such modeling concepts should be considered for general use in airbreathing hypersonic flight research vehicle simulations when developing new flight test techniques for this class of aircraft.

Nomenclature

A_c	reference inlet cowl area
A_∞	free-stream capture area
C_M	pitching moment coefficient
C_T	coefficient of thrust
CONUS	continental United States
F	propulsive force
HUD	head-up display
I_{sp}	specific impulse
JPO	Joint Program Office
L	reference length
LOX	liquid oxygen
M	Mach number or pitching moment
NASP	National Aero-Space Plane
NES	NASP Engineering Simulator
NPO	National Program Office
P	static pressure
PLA	power lever angle
$\bar{q}$	dynamic pressure
SSTO	single-stage-to-orbit (condition)
T	static temperature
α	angle of attack
Δ	difference between two values
ϕ	fuel-to-air equivalence ratio

Subscripts

∞	free-stream value
0	propulsion system station 0
1'	table value 1
2	propulsion system station 2
2'	table value 2
3	propulsion system station 3
3'	table value 3
A	axial thrust component
c	inlet cowl station
M	pitching moment
N	normal thrust component
ref	SSTO reference flight conditions
s	ramjet combustor normal shock location
T	thrust component or contribution

---

*Chief Engineer, NASP Project. Senior Member, AIAA.

Introduction

The National Aero-Space Plane (NASP), designated the X-30, is being developed to advance and validate the requisite technologies needed for airbreathing hypersonic flight within the Earth's atmosphere up to and including single-stage-to-orbit (SSTO). This includes development of major technologies such as materials and structures, controls, low-speed propulsion concepts, ramjet–scramjet airbreathing engines, cryogenic propellants, vehicle thermal management, and other aircraft subsystems. The ultimate goal is a flight vehicle to provide the final integration and validation of these technologies.

Figure 1 is an artist's concept of the X-30 aircraft. The flight vehicle is designed to be a hydrogen-powered airbreathing propulsion system aircraft capable of horizontal takeoff from a conventional runway to orbit. It will operate at high dynamic pressures to maximize airbreathing engine performance. This will result in very high airframe temperatures and heat flux rates requiring active cooling and passive thermal protection. The propulsion system includes a low-speed engine system to approximately Mach 3, ramjet operation from Mach 3 to approximately Mach 6, and a scramjet mode from Mach 6 and above. A small rocket system is planned for final orbit insertion and circularization.

The NASP research flight vehicle program consists of three distinct phases. Phase 1 was a concept evaluation period. The program is currently in the latter stages of the Phase 2 period, which includes the development and validation of the requisite technologies needed to build an actual flight vehicle. If approved in April 1993, Phase 3 will include detailed vehicle design and fabrication as well as flight test and research activities. The first flight is currently slated for late 1997 with the goal of achieving SSTO by 1999 after a two-year envelope expansion phase.

The X-30 vehicle will experience a unique flight-envelope expansion program. The program will feature a very large hypersonic flight envelope over a Mach number range to Mach 25, severe dynamic pressure loads, and an extreme aerodynamic heating environment. The challenge of this operating envelope is coupled with a unique, never-before-flown airbreathing propulsion system, complex integrated control systems for guidance and trajectory control, active cooling, and advanced structures. The entire continental United States (CONUS) will be needed to expand and explore the X-30 flight envelope. This will require complex flight test plans and range requirements for single-site operation from Edwards AFB. Flight test planning is required early in the program to develop requisite envelope expansion concepts, identify operational design features, and define range requirements.

The X-30 flight test planning and flight operational assessment has necessitated extensive development of a NASA Dryden Flight Research Facility (NASA Dryden) hypersonic real-time simulator. This is especially true in the propulsion modeling area. The large hypersonic flight envelope, multiengine modes, unique ramjet–scramjet propulsion performance, and operational requirements for flight-envelope expansion were the impetus for developing new modeling techniques and models. The Edwards government NASP flight test team consists of personnel from NASA Dryden and the Air Force Flight Test Center. Since 1986 the team has worked to expand and develop X-30 simulation technology, particularly in the modeling area. A highly flexible real-time simulation was developed with both generic unclassified hypersonic vehicle models and specific NASP configuration simulations. These have proved invaluable to the team for flight planning, design trade studies, and overall simulation technology development. Simulation work has included SSTO trajectory studies, vehicle performance and aerothermodynamics, hypersonic guidance and control techniques, and flying qualities.

This paper will present an overview of the flight planning activities to date, including a discussion of the government flight-envelope expansion concept and other design flight operational assessments. The NASA Dryden NASP real-time simulator configuration will be discussed and hypersonic flight planning simulation propulsion modeling requirements will be described. Finally, an outline of the major propulsion modeling techniques developed by the Edwards flight test team will be given with a discussion of the application value of techniques for developmental hypersonic vehicles.

X-30 Flight-Envelope Expansion Concepts

Because of the large, mostly unknown flight envelope facing the X-30 vehicle on its way to orbital flight, the basic philosophy has been to develop very conservative flight-envelope expansion concepts. This philosophy, coupled with a desire to minimize costly duplication of ground facilities, has resulted in a single test launch and recovery site concept at Edwards AFB in the Mojave Desert of California. The basic idea is to takeoff, climb subsonically, and then accelerate easterly to a conservative cruise speed outbound from Edwards. Then the vehicle will be turned around at a sufficient range, pointing in the direction of Edwards, and accelerated to the new aim envelope expansion Mach number. The aircraft will be stabilized at this test condition to obtain flight clearance data and then the engines will be throttled back or shut down to recover back at Edwards. A ground track example of this concept is outlined in Fig. 2. With the X-30 aircraft headed toward the recovery site upon reaching the new flight-envelope point, a safer recovery can be made if problems arise. This classical stabilized, incremental Mach method has been considered the safest way to carefully approach new flight conditions and–or envelope operating limits. This would include flutter boundaries,

thermal limits, engine structural or operating limits, control system critical operating points, and other conditions.

Other flight-envelope expansion concepts are being developed and evaluated. One of the basic flight test requirements is to always have the X-30 vehicle within power-off glide range of a landing site within the CONUS. Of interest then is the maximum Mach number achievable within the CONUS and the downrange and crossrange requirements. This includes assessment of available abort landing sites within the CONUS and assessments such as abort-to-orbit for recovery back to Edwards after one or more Earth orbits.

These and other operational issues are highly dependent on vehicle design, system performance, and operating limits. A key requirement for the stabilized, incremental Mach number envelope expansion method is the need to cruise the vehicle by throttling the engine thrust with either fuel and–or mass flow. This is not straightforward for an accelerator aircraft such as the X-30 airplane. To fully explore potential flight test techniques and carry out the necessary flight planning, the NASA Dryden NASP engineering simulator (NES) had to incorporate the flexible, multifaceted propulsion modeling concept described in this paper.

X-30 Engineering Simulation Description and Evolution

The NASA Dryden real-time, man-in-the-loop simulation development effort began in early 1986 with the introduction of the NASP Revised Government Baseline Vehicle. This vehicle configuration was developed at the NASA Langley Research Center (NASA Langley) from an earlier configuration design by the Dupont Aerospace Corporation of La Jolla, California. This vehicle model was followed by other hypersonic SSTO vehicle configurations including specific NASP contractor configurations.

The NES is a basic fixed-base engineering simulator made up primarily of software containing vehicle performance models. It does not include any aircraft hardware-in-the-loop or any actual vehicle cockpit layouts. These features are still in development. As seen in Fig. 3, the cockpit contains simple shuttle-type vertical tape instruments and analog round-dial instruments, with a standard center control stick and rudder pedals.

System hardware includes a pair of Gould (Encore Computer Corp., Fort Lauderdale, Florida) computers (a 32/9780 and a 32/6750) joined with shared memory. The simulator is also supported by four eight-channel strip chart recorders and an interactive user terminal. Figure 4 shows a block diagram of the simulator configuration. The entire operation can be controlled from the cockpit with both computer keyboard and switch inputs into the simulator. Digital input and output data from the models for aircraft performance and response characteristics can be recorded on a magnetic disk in real time. A hard copy of any user

terminal page is obtainable. A 25-in. monitor displays two switchable visual scenes, each generated by a Silicon Graphics IRIS 4D/80GT (Silicon Graphics, Inc., Mountain View, California) display. One scene, shown in Fig. 5, is the head-up display (HUD) superimposed over the traditional out-the-window view of Rogers Dry Lake and the surrounding local area. The other scene (Fig. 6), is an overhead view of the vehicle's ground track over five southwestern states. Two 19-in. monitors with engineering displays are driven by a MassComp 5400 (Concurrent Computer Corp., Westford, Massachusetts) workstation. Figure 7 shows one of these monitors, which provides the vehicle CONUS ground tracking and abort cardioid used to designate potential emergency landing sites. Figure 8 shows the other monitor, which is an engineering data information display. This monitor shows propulsion, aerothermodynamic heating, sonic boom overpressure, propellant, and flight Mach number and altitude information in real time along the vehicle flightpath.

Simulation software is highly modular and written in FORTRAN 77 language. Generic and vehicle-specific models include full six-degree-of-freedom oblate rotating Earth and gravity models, an atmospheric model, an aerodynamic heating model, and a sonic boom overpressure model. Other vehicle models include aerodynamics, propulsion, actuator, mass properties, controls, and guidance and navigation.

Edwards Flight Test Team Flight Planning Studies

The Edwards flight test team flight planning simulation studies have centered on envelope expansion concept development and abort scenario development–evaluation for the evolving NASP-specific configurations. This includes an operational, range, and safety assessment of the vehicle. In addition to this fundamental task, the Edwards team has supported the NASP Joint Program Office (JPO) and its government partners, the contractors, and the National Program Office (NPO) in various design trade studies and flight operational assessments of design issues.

Additionally, the team has conducted special studies in such areas as engine and airframe duty cycle definition to assess potential operating limit impacts and assist in defining vehicle life. Other special tasks have included SSTO trajectory studies and performance optimization for SSTO, vehicle heat load comparisons between the SSTO mission profile and the flight-envelope expansion missions, vehicle performance sensitivity studies, and handling qualities evaluation. Takeoff and landing performance has been evaluated from a technique and flying qualities perspective. Unpowered landing approach technique has been studied extensively.

These study results, fed directly into the vehicle design process, have greatly benefited the government flight test team in design evaluations. In addition, lessons learned from this work have been instrumental in assisting the government and contractor national team in the development of

key program planning documents. These include the Flight Test Plan and the Systems Requirements Document.

X-30 Propulsion Modeling Requirements

To properly model the X-30 propulsion system for real-time implementation on the NES simulator, several prerequisites needed definition. First, the models had to be compact and computationally efficient to conserve simulation computation time and storage capacity. Implementing and running several engine-cycle computer programs in real time was impractical. The multiengine mode nature of the airbreathing propulsion system had to be incorporated in a single model. Proper consideration was given to correct modeling of engine mode transition between the low-speed, ramjet, and scramjet engines. Additionally, a separate rocket model had to be incorporated to work with the airbreathing system across the Mach number range.

The models' propulsion performance had to be modeled as a function of flight conditions, engine power (fuel flow) settings, and atmospheric effects. This flexibility was needed to carry out various studies and develop flight test techniques for flight conditions other than those of the SSTO. Simulation models required a sufficient expansion of angle-of-attack dependency to allow for cruise and turns. Also required was a large dynamic pressure range within operating limits of the engine combustion process and cooling requirements that would allow reasonable cruise missions. Cruise is desirable at low dynamic pressure at high Mach number to minimize heat loads on the airframe and systems. An ancillary aspect of the cruise capability was the engine fuel and mass flow throttleability required to modulate thrust to cruise conditions over a large Mach number and altitude range.

Only steady-state models have been used to date, although dynamic modeling techniques are available and will be incorporated. Flight condition modeling included wide angle-of-attack, dynamic pressure, and Mach number ranges. Angle-of-sideslip effects have not been modeled yet. In addition, a flowpath liquid oxygen (LOX) augmentation capability has been modeled, but the external burning technique for base drag reduction has not been accounted for.

Separate models were developed for the external thrust and specific impulse performance, the internal engine pressure and temperature performance, and the variable engine geometry mass flow throttling. As seen in Fig. 9, separately controllable throttles were built into the cockpit for the airbreather mass and fuel flow control and the rocket throttling. The mass flow throttle controls the engine variable geometry simultaneously for all flowpaths to regulate mass flow just as the single fuel flow throttle does for the propellant. The single rocket throttle controls all rocket thruster modules. One throttle is unused. Engine fuel throttling can be performed manually by the pilot or scheduled automatically as a function of Mach number and dynamic pressure.

Expanded Propulsion Modeling Techniques

The thrust and specific impulse propulsion model for the airbreather included the three engine modes described previously. Thrust and thrust-induced pitching moment were modeled in coefficient form. They were modeled with dynamic pressure ($\bar{q}$) and a reference inlet cowl area as a function of Mach number (M), angle of attack (α), and the fuel equivalence ratio (ϕ), which is defined as the actual fuel-to-air ratio divided by the stoichiometric fuel-to-air ratio. An important aspect of this technique was that it reduces the dimensionality of the model from four (as a $f(M, \alpha, \phi, \bar{q})$) to three (as a $f(M, \alpha, \phi)$) independent variables. This makes the model implementation more compact and computationally efficient. The specific impulse (I_{sp}) over the flight envelope was modeled with reference to the SSTO specific impulse as a function of the same variables. Figure 10 shows a flowchart of the technique.

Propulsion model expansion was computed from a NASA Langley-developed computer code known as SRGULL.[1] The SRGULL program is a two-dimensional nose-to-tail ramjet–scramjet engine-cycle code. The program is made up of a combination of two-dimensional Euler inviscid flow codes for the inlet and nozzle, and a one-dimensional multi-step combustor code. A Spalding-Chi boundary-layer code was embedded to apply viscous corrections to the inviscid calculation. This was done to account for effects of skin friction and heat transfer throughout the flowpath from nose to tail, including the combustor section. An eight-specie chemical kinetics model was included to account for the effects of chemical kinetics on the combustor and nozzle flows. Inputs to SRGULL included flight conditions of Mach number, altitude or dynamic pressure, angle of attack, and power setting along with a definition of the vehicle lower fuselage geometry. This allowed for the rapid calculation of propulsion forces and moments for the model database buildup. Cowl-to-tail axial and normal thrust components as well as the thrust pitching moment contribution were computed.

Data furnished by the NASP NPO engine contractors were used to match the SSTO performance predictions and maintain model commonality over the available database variable range. The thrust, moment, and specific impulse data were converted to coefficient (thrust and moment) or SSTO-reference ratio (I_{sp}) form and implemented in the propulsion model. This database was then extrapolated with SRGULL trend analysis results to larger angle-of-attack values of approximately 15°, and to combustion limit values of equivalence ratio and dynamic pressure. Typically, most contractor-furnished models concentrated on defining the models only to low angles of attack, and high dynamic pressure and fuel-to-air equivalence ratio values corresponding

to the SSTO trajectory. Little or no data were available for cruise at low dynamic pressure and fuel-to-air equivalence ratio or at maneuvering angles of attack such as in a turn. The coefficient method advantage was that the model could be expanded easily in dynamic pressure from the high SSTO values common in the contractor models to very low dynamic pressure conditions at high altitude. Figure 11 shows an example of the model results for the axial thrust component and the specific impulse. The figure portrays excellent SRGULL results agreement within nominally ± 5 percent of the contractor propulsion data over the applicable range. It provided realistic trend data for extending the model to desired levels beyond the available database.

The coefficient of thrust and moment, and I_{sp} data were assumed constant with dynamic pressure. However, the data were a slight function of dynamic pressure, which was corrected for as a function of Mach number. The dynamic pressure correction was stored in the model as a function of Mach number and angle of attack and added to the basic model coefficients. Detailed discussions of the techniques described here and following are beyond the scope of this paper.

The LOX augmentation models were also developed for the engine to study thrust enhancement techniques. These models allow for automatic scheduling of the LOX augmentation at various preselected Mach numbers and over a range of oxidant-to-fuel ratios.

Internal Propulsion Modeling Technique

The internal engine flowpath static pressures and temperatures were modeled as a function of flight condition and power setting. This was required to track potential engine pressure and thermal operating limits during the flight planning studies. An extra value was the ability to define the engine duty cycle during the SSTO mission and typical flight test missions. The technique again involved the use of NASA Langley's SRGULL steady-state engine-cycle program to build the pressure and temperature databases as a function of engine station location. No engine flowpath or control dynamics were modeled. Figure 12 shows a summary schematic of the technique and model structure.

The variables were referenced to corresponding SSTO conditions and anchored to free-stream conditions of pressure and temperature. Engine stations modeled included the inlet throat (station 2), ramjet combustor normal shock location (station s), and combustor exit (station 3). Conditions at each engine station had to be determined beginning from free stream to model the pressure and temperature changes from station to station throughout the length of the engine. Temperature and pressure ratios were modeled specifically as a function of Mach number, angle of attack, and fuel equivalence ratio in a range corresponding to that of the external propulsion model. Ratios were assumed constant with dynamic pressure.

The opinion of the Edwards team is that the modeling techniques discussed previously for expanded and internal propulsion modeling will be required for developmental flight test planning for this class of hypersonic airbreathing vehicles. They are generally applicable to a wide range of such vehicles for modeling vehicle performance in either the batch or real-time simulation modes.

Thrust Modulation Modeling Techniques

As discussed earlier, the current government NASP flight-envelope expansion concept is based on the stabilized, incremental Mach number expansion technique. In turn, this technique relies on the ability to throttle or modulate the airbreather engine thrust to a stabilized cruise condition. This is particularly difficult for the NASP SSTO hypersonic class of aircraft that are designed to be high-speed accelerators to orbit with limited cruise design capability. Fuel flow throttling alone won't allow sufficient thrust modulation because of practical engine operating limitations such as a minimum combustor pressure limit for flame holding and combustion. This is particularly critical in the ramjet mode. Additionally, the scramjet typically has minimum engine cooling requirements using the circulated fuel for cooling the entire airframe structure. Once a minimum fuel flow condition is reached the aircraft normally would still have a large thrust residual. The only other way to reduce the thrust level is by throttling-down the inlet mass flow using available variable engine geometry.

This led to the idea for developing an engine mass flow throttling model using the SRGULL program to calculate the modulation of engine mass flow. The thrust coefficient components and pitching moment contribution described in an earlier section were calculated with reduced mass flow and modeled as a function of the same variables described previously. The objective was to model the mass flow throttling so that an efficient, linear variation of thrust could be obtained using a separate cockpit throttle from the one used to modulate fuel flow. A schematic of the control schedule concept is shown in Fig. 13. Development and implementation of this control schedule allowed for vehicle stabilization over virtually its entire flight envelope. This was done without compromising minimum combustor pressure limits or engine minimum cooling fuel flow limits. Also modeled and compensated for are the inlet supersonic flow unstart limits, especially for the ramjet mode. The geometry control scheduling is automatic and programmed as a function of the mass flow throttle lever angle.

Figure 14 shows the axial coefficient of thrust and specific impulse as a function of percent mass flow throttle setting for the ramjet at Mach 4 and for the scramjet at Mach 6. The data shown are for a given dynamic pressure, angle of attack, and fuel-to-air equivalence ratio flight condition. The figure clearly illustrates the technique's effectiveness in producing a large, linear thrust modulation capability. The developed

throttling technique's advantage is that it allows easy tailoring of the engine thrust and–or specific impulse response for any desirable control characteristic. This technique has excellent utility for flight studies and other flight applications such as hypersonic cruiser aircraft.

Concluding Remarks

The Edwards AFB government flight test team of the National Aero-Space Plane Program has successfully developed several innovative propulsion modeling techniques for real-time simulation. These techniques were required to allow maximum flexibility in ongoing flight planning and flight study activities to develop flight-envelope expansion techniques and to address flight operational issues. Technique development has included a thrust and specific impulse model expanded beyond the normal SSTO flight conditions to allow for cruise and maneuvering flight. In addition, an internal flowpath propulsion model was developed to better monitor engine operating limits and determine duty cycles. A powerful thrust modulation technique was developed through the control of mass flow and fuel flow. This has allowed a greatly expanded cruise envelope capability, which is needed for some of the flight-envelope expansion concepts being developed.

These modeling techniques have been applied to the NASA Dryden Flight Research Facility National Aero-Space Plane (NASP) Engineering Simulator, yielding high-fidelity simplified, and computationally fast real-time simulation models. These techniques are generally applicable to a wide range of airbreathing hypersonic vehicles for modeling vehicle performance in the batch or real-time simulation modes. It is the opinion of the Edwards team that these modeling techniques will be required for developmental flight test planning and for other flight applications such as hypersonic cruiser aircraft.

References

[1] Pinckney, S. Zane, and Walton, James T., *Program SRGULL: An Advanced Engineering Model for the Prediction of Airframe-Integrated Subsonic/Supersonic Hydrogen Combustion Ramjet Cycle Performance*, NASP TM-1120, 1991.
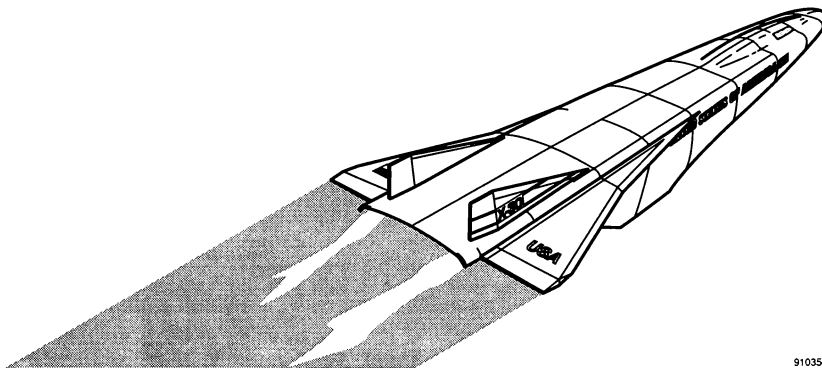
910354

Fig. 1 Artist's concept of the X-30 vehicle.
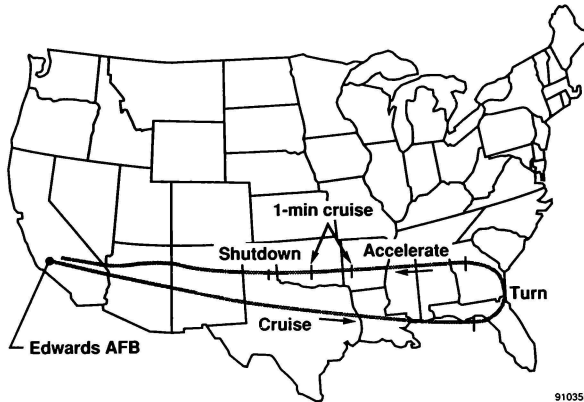
Fig. 2 Typical envelope expansion flight test mission.
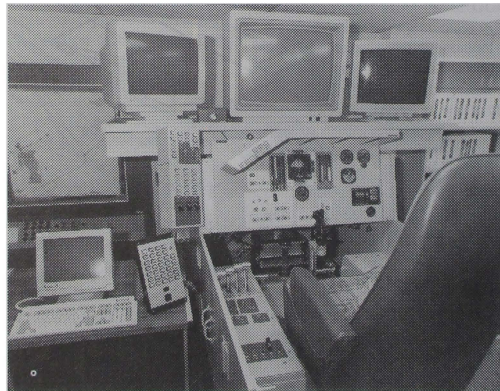
EC 90 236-4

Fig. 3 NES cockpit layout.
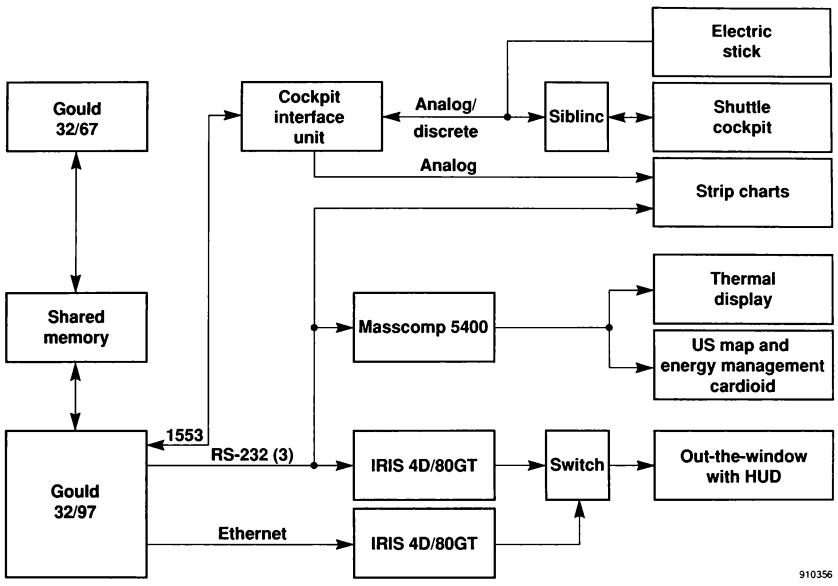
Fig. 4 Current simulation configuration.
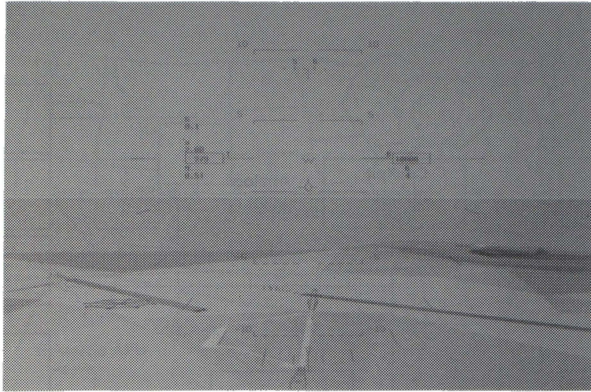
EC 91 217-1

Fig. 5 NES out-the-window scene.
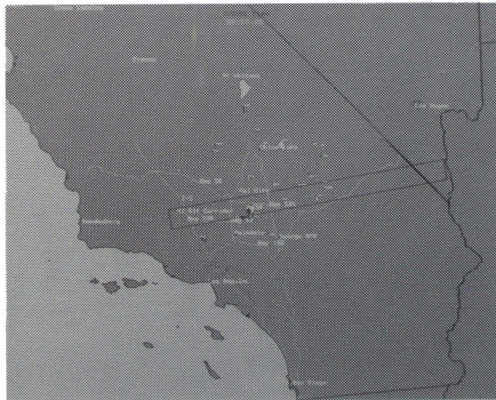
EC 91 217-3

Fig. 6 Local area display.
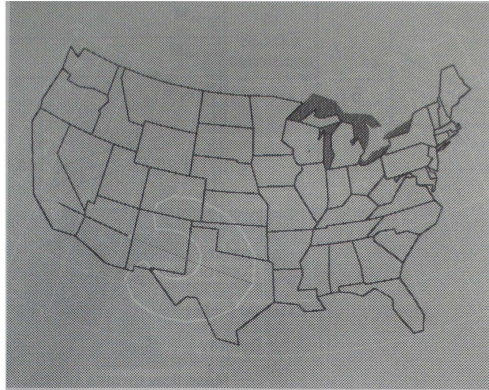
EC 90 227-5

Fig. 7 CONUS map with recovery cardioid.
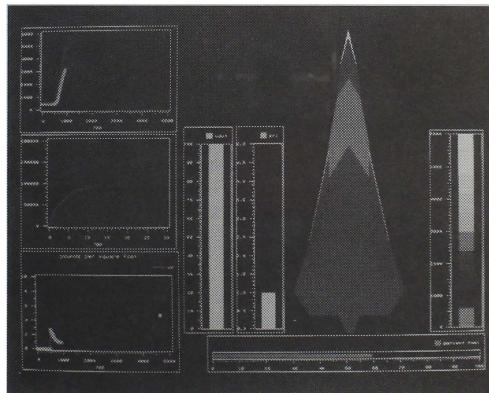
EC 90 227-4

Fig. 8 Engineering data display.

Control stick

External rocket

Dummy

Mass flow throttle

Fuel flow throttle
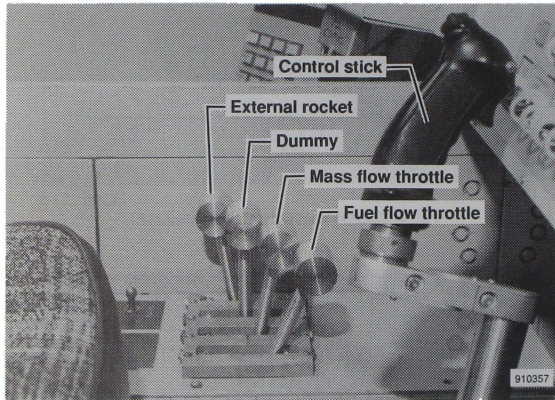
910357

EC 90 236-2

Fig. 9 NES engine throttles.

220
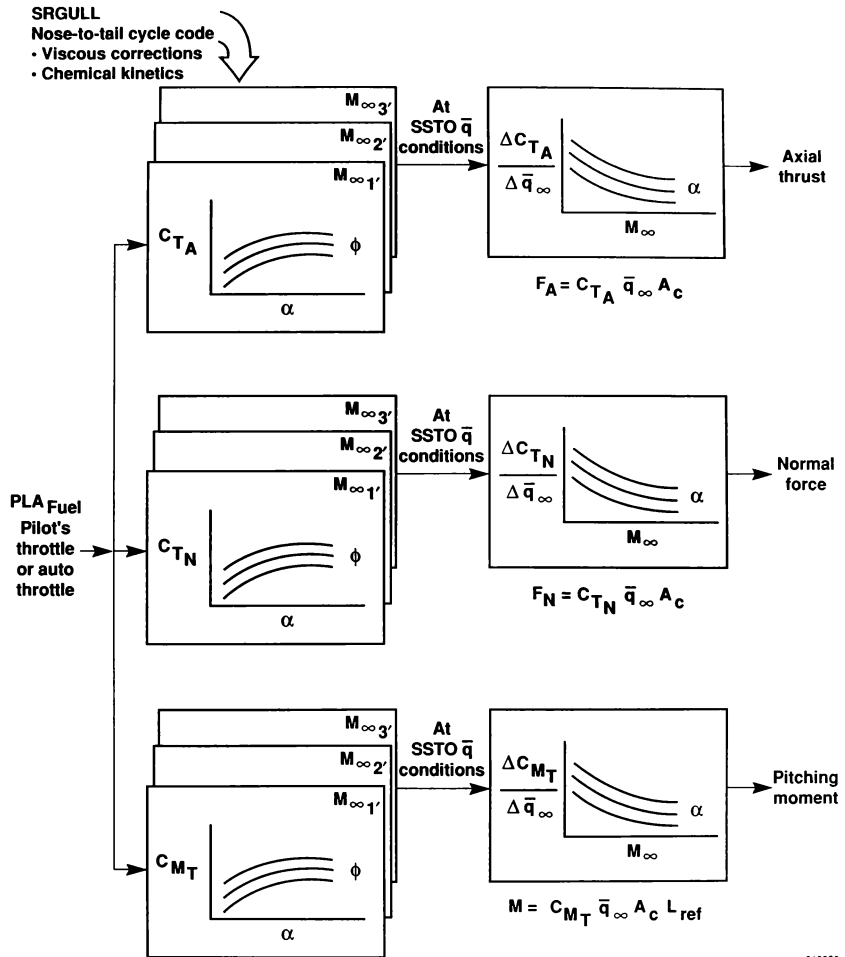
Fig. 10 Scramjet propulsion modeling technique.

Fig. 11 Typical scramjet propulsion model, results from SRGULL and simulation; $\phi = 1.0$.

Fig. 12 Internal propulsion modeling technique.
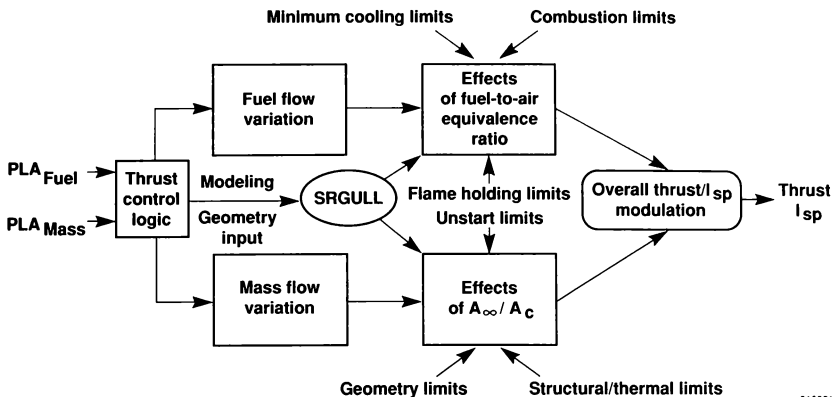
Fig. 13 Propulsion throttling concept using SRGULL model calculations.
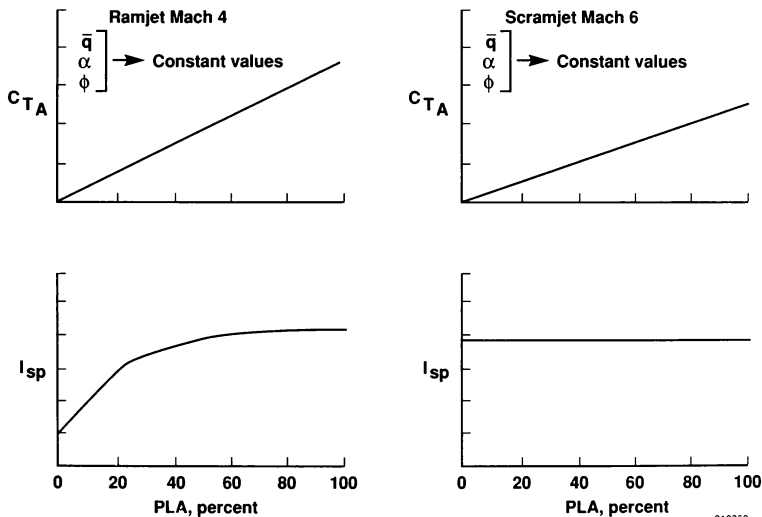
Fig. 14 Mass flow thrust modulation effectiveness.

DYNAMICS OF AERO-DRIVEN BODIES WITH COLLISIONS AND SLIDING CONTACT

D. T. Wang*, G. A. Nystrom†,and C. K. Lombard‡
PEDA Corporation, Palo Alto, California 94303

Abstract

The relations governing sabot-discarding motion with either collisions or sliding contact are obtained. Numerical solutions are presented to illustrate the two-dimensional behavior of sabot components under the influence of aerodynamic forces and contact with the nearby projectile. The number of collisions is found to be a strong function of initial conditions and the collision coefficients.

Nomenclature

A	inertial reference frame
F_i	resultant of contact and distance forces acting on particle i
F_x, F_y	aerodynamic loads on sabot in x and y directions
F_h	contact force between sabot and projectile
G	gap between sabot rear and projectile
G_r	mathematical functions, $r = 1, 2$
H, H'	contact points of sabot and projectile
I_r	generalized impulse, $r = 1, ..., 12$
J_c	moment of inertia of sabot about mass center for 2-D calculation
J_{S1}, J_{S2}, J_{S3}	central principal moments of inertia of sabot
J_{P1}, J_{P2}, J_{P3}	central principal moments of intertia of projectile
K, K_S, K_P	kinetic energy and kinetic energies of sabot and projectile
L	sabot length
M_o	freestream conditions
N	cartesian coordinate system fixed in projectile
P	projectile
P_r	generalized momentum, $r = 1, ..., 12$
P^*	mass center of the projectile
R, R_1, R_2, R_3	total contact force exerted on sabot by projectile and its 3 Cartesian components
S	sabot
S^*	mass center of sabot

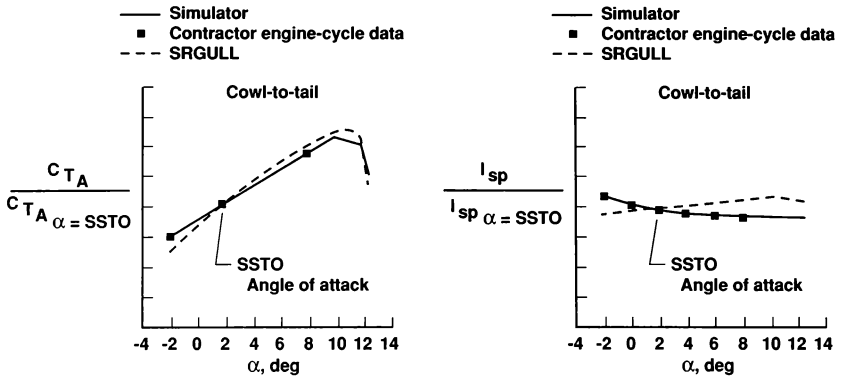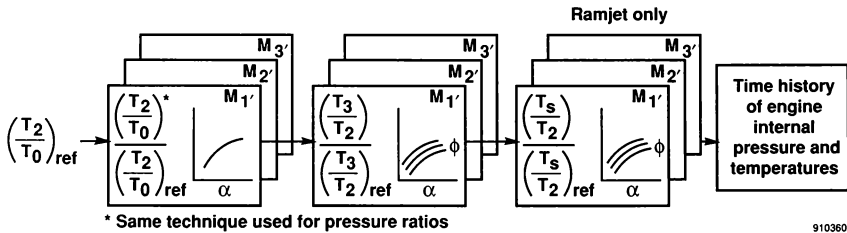|	
S_1, S_2, S_3	impulses in three Cartesian components
T	A plane which is tangent to the surface of sabot and projectile at points of contact
T_S^*, T_P^*	inertia torques of sabot and projectile
$^A\mathbf{V}^{S^*}$, $^A\mathbf{V}^{P^*}$	velocities of mass centers of sabot and projectile in A
$^A\mathbf{V}^H$, $^A\mathbf{V}^{H'}$	velocities of contact points H and H' in A
V_A, V_S	velocities of approach and separation
$^AV_r^i$, $^AV_r^H$, $^AV_r^{H'}$	rth partial velocities of particle i as well as points H and H', $r = 1, ..., 12$
a, b, c, d, e, f	Cartesian components of position vectors $\mathbf{r}_1$ and $\mathbf{r}_2$
$\mathbf{a}^{S^*}$, $\mathbf{a}^{P^*}$	accelerations of mass centers of sabot and projectile
a_x, a_y	accelerations of center of gravity of sabot in x and y directions
h	coefficient of restitution
m_i, m_s, m_p	masses of particle i, sabot, and projectile
$\mathbf{n}$	a unit vector perpendicular to T
$\mathbf{n}_1$, $\mathbf{n}_2$, $\mathbf{n}_3$	Cartesian unit vectors fixed in A
$\mathbf{n}_1'$, $\mathbf{n}_2'$, $\mathbf{n}_3'$	Cartesian unit vectors fixed in projectile
$\mathbf{r}_1$, $\mathbf{r}_2$	position vectors from mass center of sabot to point H and from mass center of projectile to point H'
t_1, t_2	initial and final instants of time of collision
u_r	generalized speeds, $r = 1, ..., 12$
v_x, v_y	velocity of center of gravity of sabot in x and y directions
$^A\omega^S$, $^A\omega^P$	angular velocities of sabot and projectile in A
$^A\omega_r^S$, $^A\omega_r^P$	rth partial angular velocities of sabot and projectile in A
μ, μ'	coefficients of static and kinetic friction for sabot and projectile
ν	number of particles in a system
ς	normal impulse
τ	tangential impulse
α	angular acceleration of sabot

---

* Research Engineer, Member AIAA
† Currently in Failure Analysis Associates
‡ President, Member AIAA

Introduction

A sabot is required to launch the antiarmor, kinetic energy projectile. Within the bore, the sabot provides gas sealing and structural support. However, once free of the gun tube, the sabot must be discarded in order to permit unconstrained, low drag flight to the target[1]. Typically, the sabot is divided into three or more components along

axial planes as shown in figure 1. At separation from the gun tube, the sabot and projectile are in direct mechanical contact. Elastic decompression, spin, and aerodynamic loadings act to lift the sabot components away from the projectile. However, prior to separation mechanical interaction may persist if the components pivot about a point of contact which we call sliding contact. Alternatively, the sabot components may initially break contact only to reimpinge on the projectile at a later stage in the discard which we call collision. Asymmetries in the contact process, e.g., contact between the projectile and only one of the sabot components, can adversely affect the projectile trajectory.

The present paper reports on part of an on-going computational analysis of nonlinear projectile motion and its interaction with its sabot during discard. It mainly deals with the rigid body dynamics of the various components during free-flight, sliding contact, and collision.

Mathematical Model

The separation of a sabot from the projectile could consist of four phases: initial contact, sliding contact, free flight, and collision as shown in figure 2 . We shall first derive the governing relations applicable for sliding contact and free flight. Then the relations applicable for collision will be derived.

Sliding Contact Model

At the beginning, the sabot is in initial contact with the projectile. Once free of the gun tube, the sabot moves away from the projectile under aerodynamic loadings. Sliding contact could occur at this moment. The equations of motion for sliding contact are derived as follows:

$$F_x - \mu' F_h = m a_x \qquad (1)$$

$$F_y + F_h = m a_y \qquad (2)$$

$$F_h l_x - \mu' F_h l_y + \Gamma = J_c \alpha \qquad (3)$$

where μ' is kinetic coefficient of friction; F_h is contact force in normal direction; F_x, F_y, and Γ correspond to aerodynamic loadings of two force components in x and y directions and a torque in z direction; a_x, a_y, and α are accelerations of center of gravity of sabot in x and y directions and angular accelation in z direction; m is mass of sabot; J_c is moment of inertia of the sabot about its center of gravity. Figure 3 shows the sketch of model in which M_o means the free stream conditions. The projectile is assumed to have a constant velocity and a flat surface. The direction of frictional force, $\mu' F_h$, is always opposite to the direction of motion of sabot's contact point.

The constraint equation for sliding contact can be expressed as

$$v_h \cdot n = 0 \qquad (4)$$

where v_h is velocity of sabot at point H and n is a unit vector in y direction. From kinematics the velocity of center of gravity of sabot in normal direction can be obtained as $v_y = -\omega l_x$, where ω is angular velocity of sabot. The

acceleration of sabot in normal direction is

$$a_y = -\alpha l_x \qquad (5)$$

Four unknowns a_x, a_y, α, and F_h are determined by solving eqns. (1), (2), (3), and (5) simultaneously. F_h greater than zero corresponds to case of contact and equal to zero corresponds to case of separation(free flight).

Free Flight

For cases of free flight, contact force F_h is equal to zero. Therefore, eqns. (1), (2), and (3) reduce to the three fundamental equations of motion with F_x, F_y, and Γ only.

Collision Model

During collision of sabot and projectile, it is assumed that the collision takes place over a very small time increment such that the velocities change during collision event, but not the geometries. This assumption will allow us to neglect details involved in the actual collision process. S^* and P^* are the mass centers of sabot, S, and projectile, P, respectively. Position vector $\mathbf{r}_1$ is a vector from S^* to the contact point H of sabot and $\mathbf{r}_2$ is a position vector from P^* to the contact point H' of projectile. $\mathbf{n}_1$, $\mathbf{n}_2$, $\mathbf{n}_3$ is a right-handed set of mutually perpendicular unit vectors of inertia reference frame A fixed in earth and $\mathbf{n}'_1$, $\mathbf{n}'_2$, $\mathbf{n}'_3$ is a right-handed set of mutually perpendicular unit vectors of coordinate system N fixed in projectile as shown in figure 4.

The velocity $^A\mathbf{V}^{S^*}$ of S^* and angular velocity $^A\omega^S$ of S in referance frame A can be expressed in terms of generalized speeds u_i, $i = 1,..,6$, as

$$^A\mathbf{V}^{S^*} = \sum_{i=1}^{3} u_i \mathbf{n}_i \quad \text{and} \quad ^A\omega^S = \sum_{i=1}^{3} u_{i+3} \mathbf{n}_i$$

Likewise, the velocity $^A\mathbf{V}^{P^*}$ of P^* and angular velocity $^A\omega^P$ of P in reference frame A can be expressed in terms of generalized speeds u_i, $i = 7,..,12$, as

$$^A\mathbf{V}^{P^*} = \sum_{i=1}^{3} u_{i+6} \mathbf{n}_i \quad \text{and} \quad ^A\omega^P = \sum_{i=1}^{3} u_{i+9} \mathbf{n}_i$$

The kinetic energy of S and P is given by

$$
\begin{aligned}
K &= K_S + K_P \\
&= \frac{1}{2} m_s \sum_{i=1}^{3} u_i^2 + \frac{1}{2} \sum_{i=1}^{3} J_{s_i} u_{i+3}^2 \\
&\quad + \frac{1}{2} m_p \sum_{i=7}^{9} u_i^2 + \frac{1}{2} \sum_{i=1}^{3} J_{p_i} u_{i+9}^2
\end{aligned}
$$

where m_s and m_p are the masses of S and P, respectively; J_{s_1}, J_{s_2}, J_{s_3} are central principal moments of inertia of S

and J_{p_1}, J_{p_2}, J_{p_3} are central principal moments of inertia of P.

The generalized momentum P_r is defined as $P_r \equiv \sum_{i=1}^{\nu} m_i\ {}^A\mathbf{V}_r^i \cdot {}^A\mathbf{V}^i$, where ${}^A\mathbf{V}_r^i$ is the rth partial velocity of a generic particle i in A; ${}^A\mathbf{V}^i$ is the velocity of i in A; m_i is the mass of i; and ν is the number of particles. P_r could be simplified by the equation $P_r = \frac{\partial K}{\partial u_r}$ $(r = 1, ..., 12)$. Therefore,

$$P_r = m_s u_r \quad (r = 1, 2, 3)$$
$$P_r = m_p u_r \quad (r = 7, 8, 9)$$
$$P_4 = J_{s_1} u_4 \ , \quad P_5 = J_{s_2} u_5 \ , \quad P_6 = J_{s_3} u_6$$
$$P_{10} = J_{p_1} u_{10} \ , \quad P_{11} = J_{p_2} u_{11} \ , \quad P_{12} = J_{p_3} u_{12}$$

The partial velocities of H obtained by ${}^A\mathbf{V}_r^H = \frac{\partial {}^A\mathbf{V}^H}{\partial u_r}$ and by kinematics, are

$$\begin{aligned}
{}^A\mathbf{V}_r^H &= \mathbf{n}_r \quad (r = 1, 2, 3) \\
{}^A\mathbf{V}_4^H &= -c\mathbf{n}_2 + b\mathbf{n}_3 \ , \quad {}^A\mathbf{V}_5^H = c\mathbf{n}_1 - a\mathbf{n}_3 \ , \\
{}^A\mathbf{V}_6^H &= -b\mathbf{n}_1 + a\mathbf{n}_2 \ , \\
{}^A\mathbf{V}_r^H &= 0 \quad (r = 7, 8, ..., 12)
\end{aligned}$$

The partial velocities of H' are

$$\begin{aligned}
{}^A\mathbf{V}_r^{H'} &= 0 \quad (r = 1, 2, ..., 6) \\
{}^A\mathbf{V}_r^{H'} &= \mathbf{n}_{r-6} \quad (r = 7, 8, 9) \\
{}^A\mathbf{V}_{10}^{H'} &= -f\mathbf{n}_2 + e\mathbf{n}_3 \ , \quad {}^A\mathbf{V}_{11}^{H'} = f\mathbf{n}_1 - d\mathbf{n}_3 \ , \\
{}^A\mathbf{V}_{12}^{H'} &= -e\mathbf{n}_1 + d\mathbf{n}_2
\end{aligned}$$

Letting $\mathbf{R} \equiv \sum_{i=1}^{3} R_i \mathbf{n}_i$ be the force exerted on S by P at their point of contact during the time increment from t_1 to t_2, and defining S_i such that the following relation exists

$$\int_{t_1}^{t_2} \mathbf{R}\,dt \equiv \sum_{i=1}^{3} S_i \mathbf{n}_i'$$

Figure 5 shows the relationship between coordinate systems A and N. The transformation matrix for these two systems is a multiplication of two 3×3 matrices,

$$\begin{pmatrix} \cos\phi & 0 & -\sin\phi \\ 0 & 1 & 0 \\ \sin\phi & 0 & \cos\phi \end{pmatrix} \text{ and } \begin{pmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix}. \text{ This re-}$$

sults in the following:

$$\mathbf{n}_1' = \cos\theta\cos\phi\mathbf{n}_1 + \cos\phi\sin\theta\mathbf{n}_2 - \sin\phi\mathbf{n}_3$$
$$\mathbf{n}_2' = -\sin\theta\mathbf{n}_1 + \cos\theta\mathbf{n}_2$$
$$\mathbf{n}_3' = \cos\theta\sin\phi\mathbf{n}_1 + \sin\theta\sin\phi\mathbf{n}_2 + \cos\phi\mathbf{n}_3$$

The generalized impulse I_r is defined by $I_r \equiv \sum_{i=1}^{\nu} V_r^i(t_1) \cdot \int_{t_1}^{t_2} F_i\,dt$, where F_i is the resultant of contact and distance forces acting on particle i; and t_1 t_2 are the initial and final instants of time for the collision. The generalized impulse is given by

$$I_r = {}^A\mathbf{V}_r^H \cdot \int_{t_1}^{t_2} \mathbf{R}\,dt + {}^A\mathbf{V}_r^{H'} \cdot \int_{t_1}^{t_2} (-\mathbf{R})\,dt$$

which is equal to the difference between generalized momentum at t_2 and generalized momentum at t_1, i.e. $I_r = P_r(t_2) - P_r(t_1)$. This would result in the following twelve equations

$$\mathbf{n}_k \cdot \left(\sum_{i=1}^{3} S_i \mathbf{n}_i' \right) = m_s\,du_k$$

$$(-c\mathbf{n}_2 + b\mathbf{n}_3) \cdot \left(\sum_{i=1}^{3} S_i \mathbf{n}_i' \right) = J_{s_1}\,du_k$$

$$(c\mathbf{n}_1 - a\mathbf{n}_3) \cdot \left(\sum_{i=1}^{3} S_i \mathbf{n}_i' \right) = J_{s_2}\,du_k$$

$$(-b\mathbf{n}_1 + a\mathbf{n}_2) \cdot \left(\sum_{i=1}^{3} S_i \mathbf{n}_i' \right) = J_{s_3}\,du_k$$

$$\mathbf{n}_{k-6} \cdot \left(-\sum_{i=1}^{3} S_i \mathbf{n}_i' \right) = m_p\,du_k$$

$$(-f\mathbf{n}_2 + e\mathbf{n}_3) \cdot \left(-\sum_{i=1}^{3} S_i \mathbf{n}_i' \right) = J_{p_1}\,du_k$$

$$(f\mathbf{n}_1 - d\mathbf{n}_3) \cdot \left(-\sum_{i=1}^{3} S_i \mathbf{n}_i' \right) = J_{p_2}\,du_k$$

$$(-e\mathbf{n}_1 + d\mathbf{n}_2) \cdot \left(-\sum_{i=1}^{3} S_i \mathbf{n}_i' \right) = J_{p_3}\,du_k$$

where $du_k = u_k(t_2) - u_k(t_1)$, k goes from 1 to 3 for the first equation, from 4 to 6 for the second to the fourth equations, from 7 to 9 for the fifth equation, and from 10 to 12 for the last three equations.

The velocity of approach, $\mathbf{V}_a$ is given by $\mathbf{V}_a \equiv {}^A\mathbf{V}^H(t_1) - {}^A\mathbf{V}^{H'}(t_1)$ and the velocity of separation is given by $\mathbf{V}_s \equiv {}^A\mathbf{V}^H(t_2) - {}^A\mathbf{V}^{H'}(t_2)$ Applying the coefficient of restitution h, the following equation exists.

$$\mathbf{V}_s \cdot \mathbf{n}_2' = -h(\mathbf{V}_a \cdot \mathbf{n}_2') \tag{6}$$

The assumption made here is that the normal components of $\mathbf{V}_a$ and $\mathbf{V}_s$ have opposite directions, while the magnitude of the normal component of $\mathbf{V}_s$ is proportional to that of $\mathbf{V}_a$, the constant of proportionality being a coefficient of restitution whose value depends on the material properties.

Generally speaking, the contact of sabot and projectile is such a hit that no slippage occurs. However, sometimes slipping does happen. One way to determine whether it happens or not is to assume the existence of a friction coefficient. The assumption involves both the tangential component of $\mathbf{V}_s$ and the integration of $\mathbf{R}$ with respect to time from t_1 to t_2 as shown in figure 6. The latter is resolved into two components, one normal to the plane T which is the plane that is tangent to the surfaces of S and P at their point of contact, denoted by ς, and called the normal impulse, the other is parallel to T, denoted by τ, and called the tangential impulse. The assumption is that if and only if the inequality $\mid \tau \mid < \mu \mid \varsigma \mid$ is satisfied, where μ is the coefficient of static friction for S and P, then there is no slipping at t_2. The no-slip condition may be stated as

$$\mathbf{n}_2' \times (\mathbf{V}_s \times \mathbf{n}_2') = 0$$

Define G_1 and G_2 such that $\mathbf{n}_2' \times (\mathbf{V}_s \times \mathbf{n}_2') \equiv G_1 \mathbf{n}_1' + G_2 \mathbf{n}_3' = 0$

$$\text{therefore,} \qquad G_1 = 0 \qquad (7)$$

$$G_2 = 0 \qquad (8)$$

If $\mid \tau \mid \geq \mu \mid \varsigma \mid$, there is slipping at t_2 and equation $\tau = -\mu' \mid \varsigma \mid \frac{\mathbf{n}_2' \times (\mathbf{V}_s \times \mathbf{n}_2')}{\mid \mathbf{n}_2' \times (\mathbf{V}_s \times \mathbf{n}_2') \mid}$ applies, where μ' is the coefficient of kinetic friction for S and P. This equation can be solved to obtain the following two formulas.

$$S_1 = -\mu' \mid S_2 \mid \frac{G_1}{(G_1^2 + G_2^2)^{\frac{1}{2}}} \qquad (9)$$

$$S_3 = -\mu' \mid S_2 \mid \frac{G_2}{(G_1^2 + G_2^2)^{\frac{1}{2}}} \qquad (10)$$

These fifteen equations and two slip inequalities can be solved simultaneously to obtain $u_r(t_2)$ $(r = 1, ..., 12)$, S_1, S_2 and S_3 given the state just prior to collision (t_1).

After this work was completed, Kane and Levinson published their derivation of two-body collision problem[2]. Since their derivation and ours were both based on the Kane and Levinson's textbook[3], it is not surprising that the two derivations are similiar.

It is of worth to mention the differences between sliding contact and collision. When the gap is closed, the normal velocity of the contact point is examined. If it is zero, sliding-contact calculations are performed, while if it is nonzero, collision calculations are performed. Sliding contact occurs for a measurable period of time, while collision time is infinitesimal. Sliding and collision calculations will be illustrated in the following section.

Illustrative Solutions

A computer program was developed which obtains numerical solutions to the two-dimensional relations presented earlier. For these illustrative calculations, the real sabot geometry was simplified by modelling the sabot as a two-dimensional slender rectangle and the projectile as a very long rectangle.

The numerical integration of the free-flight relations was straight-forward, as was the numerical solution of the relations governing the instant of impact. The most challenging part of this computer program was the logic needed to decide the time and conditions at the instant of each contact. If contact were possible at only one point, one could easily determine when the gap at that one point is zero (a negative gap would correspond to forbidden interpenetration). But when contact can occur simultaneously at several points, the logic becomes much more complicated. Our algorithm handles multiple contacts separated by arbitrary time intervals. The case of simultaneous contact at several points will be illustrated by some of the solutions below.

For the illustrative solutions which follow, the sabot is a rectangle with the following properties: length = .356 m, width = .051 m, mass = 0.45 N sec^2/m, and polar inertia = 4.8E-3 N*m*sec^2. The projectile parameters are: length = large, mass = 4.5 N sec^2/m, and polar inertia = 5.0E-1 N*m*sec^2. For all calculations, the projectile is initially stationary and centered at the origin of coordinates. Table 1 lists all the other input parameters.

The first series of calculations illustrates the effect of collision parameters on the sabot motion. For this series, the slender sabot is heading towards the projectile at an angle of 45 degrees (-0.7854 radians in Table 1), and there are no aerodynamic forces.

Case 1A is the baseline; the friction coefficients (both static and kinetic) have a value of 0.3, and the coefficient of restitution has a value of 0.8. Table 2 summarizes the input and calculated time-history. We observe that since aerodynamic forces are absent, during free-flight the velocities are constant. During each instant of contact, the velocities take a step jump, in accord with the relations discussed earlier. For Case 1A, corner 4 contacts at 0.330 seconds. The normal velocity of contact point to projectile surface is not equal to zero and collision occurs. The normal and tangential impulse are related by the specified friction coefficient (0.3). Also note that the local velocities of approach and separation are related by the specified coefficient of restitution (0.8). Finally, note that impact has converted most of the linear momentum into angular momentum. Figure 7 illustrates the sabot motion for Case 1A. It shows the vertical position and orientation as a function of time.

Case 1B is exactly the same as case 1A, except that the friction coefficients were changed from 0.30 to 0.03. As summarized on Table 1, the reduced friction coefficient is responsible for a second collision. Evaluating the effect of the reduced friction on the first collision, one finds that the reduced friction reduces the normal impulse. The normal impulse is not sufficient to reverse the downward momen-

tum of the sabot, so a second collision takes place. After the second collision, the angular velocity is relatively small, while the vertical velocity is relatively large (compared to case 1A, which involved only one collision.)

Case 1C is exactly the same as case 1A except that the coefficient of restitution was decreased from 0.80 to 0.25. Table 1 shows that, as one would expect, the momentum transfer is quite sensitive to the coefficient of restitution. When the coefficient of restitution was high (case 1A), the vertical momentum was converted to angular momentum. When the coefficient of restitution is low (Case 1C), the vertical momentum is absorbed by the relatively inelastic collisions. An interesting point is that for the lower coefficient, the number of collisions increases from 1 to 3. The collision process normally consists of the sequence of right corner collision, left corner collision, right corner collision, left corner collision....until the sabot clears the projectile. For a smaller coefficient of restitution, each collision transfers less momentum. Therefore, the total number of collisions required to reverse the sabot's momentum is also increased. We also note that each collision typically transfers less momentum than preceding collisions for that time-history. For example, for Case 1C the magnitude of normal impact is: 0.320 N sec at time 0.330 sec; 0.131 N sec at time 0.597 sec; and 0.0218 N sec. at time 1.94 sec.

If the coefficient of restitution is small enough, after a dozen collisions, the collisions have the value of our computer's roundoff. Such a situation corresponds to the two bodies being in close proximity and touching gently. For our application, contact time will be small. If one wanted to model slow and gentle contact, then the analysis should account for the structural deformation and the finite contact time.

The second series of solutions evaluates simultaneous contact at several points in the absence of aerodynamic forces. For these cases, the sabot and projectile hit broadside at a very small angle: +1.E-4, 0.0, and -1.E-4 radians for cases 2A, 2B, and 2C respectively. The normal velocities of contact points of these cases are not equal to zero. Therefore, they are all the cases of collision. Table 1 summarizes the results. For all three cases, three contacts were required for the sabot to clear the projectile. For cases 2A and 2C, the small approach angle determines which corner makes the first contact. After that first contact, the two cases possess the expected symmetry.

Case 2B is most interesting. In that case, the sabot was intended to contact the projectile simultaneously at two corners. But random round-off error determines which corner will contact first. After that random choice is made, the solution is essentially the same as either Case 2A or 2C. This is very similar to what one would obtain experimentally if one tried to drop one flat rigid body on top of another. One would find that as hard as one tried to obtain perfectly simultaneous contact along the whole surface, small imperfections in the geometry would force one particular point to make first contact.

The third series of calculations illustrate sliding contact. Sabot aerodynamic loads were applied approximately as a function of the sabot orientation and freestream conditions. Table 3 shows the aerodynamic loads F_x, F_y, and T_{ok} corresponding to forces in the x and y directions as well as moment in the z direction, respectively. In the near future, aerodynamic loads will be obtained from our finite-difference multi-grid time-accurate Navier Stokes code[4].

Immediately after the sabot and projectile come out of the gun tube, they are in contact. For case 3, the initial sabot velocity relative to the projectile was zero. The initial position corresponds to contact at the sabot's rear corner and a 10^{-7} radian orientation clockwise of the projectile. The applied aerodynamic loads tend to push the sabot away from the projectile and increase its clockwise rotation.

Table 3 shows the calculated time history and figure 8 illustrates the trend of motion of sabot. For the first four time steps contact is maintained. The pivot point moves downstream at the same time that the sabot increase its clockwise rotation. At time $4.8^* 10^{-7}$ sec, the contact force F_h has become zero. For the remaining time steps, the sabot is in free flight. The time step of sliding motion 1.2^*10^{-7} seconds is determined by an aerodynamic condition (Courant number = 1.0). Because of the difference in the orders of parameters, the size and orientation of sabot are not scaled in figure 8.

The appendix lists the definition of variables of time histories in tables 2 and 3.

Concluding Remarks

The equations governing two-body collision and sliding contact are derived. These relations are illustrated by three series of examples including a coupled aerodynamics and rigid body dynamics case. The simulation of sabot motion in the real flight could be made in our computer program by giving the flight aerodynamic loads and necessary inputs of dynamics parameters.

References

1. Schmidt, E., Disturbance to the Launch of Fin-Stabilized Projectiles, AIAA-80-1589R, *Journal of Spacecraft and Rockets*, Vol. 19, Number 1, Jan. 1982.

2. Kane, T.R. and Levinson, D.A., An Explicit Solution of the General Two-Body Collision Problem, *Computational Mechanics*, 2., 1987, pp. 75-87.

3. Kane, T.R. and Levinson, D.A., *Dynamics: Theory and Applications*, McGraw-Hill, New York, 1985.

4. Hong, S.K., Nystrom, G.A., Wang, D.T., Bardina, J. and Lombard, C.K., Simulation of 3-D Jet-Interaction Flowfields with CSCM on Multiple Grids, AIAA-89-2552, July, 1989.

Input

Q(l), l = 1,..,6   Initial geometry. The first three coordinates locate the sabot location and orientation, while the last three coordinates locate the projectile location and orientation. For more details, see the output discussion.

U(l), l = 1,...,6   Initial Velocities. These are time derivatives of the initial geometry coordinates, Q(l). For more details, see the output discussion.

THO = Density of sabot

PM = Projectile mass.

SJ = Sabot polar moment of inertia.

PJ = Projectile polar moment of inertia.

TOTIME = Total time of integration required.

TIMEIV = Time interval for performing the time integrations.

R(l), l = 1,..,4   Radii from the sabot center of gravity to the sabot's four corners.

COES = Static friction coefficient. Its value is expected to be in the order of 0.3.

COEK = Kinematic friction coefficient. Its value is expected to be in the order of 0.2.

THETA(l), l = 1,..,4   Angular orientation of the four corners when the sabot is in the reference orientation with Q(3) = 0.

COER = Coefficient of restitution. Its value must lie between 0. and 1.

RR = Projectile radius.

FX = Aerodynamic loads on sabot in x direction

FY = Aerodynamic loads on sabot in y direction

TOK = Aerodynamic loads on sabot in z direction

Output During Free Flight

TIME = Time since the inital conditions were specified.

GAP = Smallest gap between the sabot and projectile.

DXSABOT = X-coordinate of the sabot center of gravity.

DYSABOT = Y-coordinate of the sabot center of gravity.

DROTSABOT = Anti-clockwise rotation of the sabot.

DXPROJ = X-coordinate of the projectile center of gravity.

DYPROJ = Y-coordinate of the projectile center of gravity.

DROTPROJ = Anti-clockwise rotation of the projectile.

UXSABOT = X-velocity of the sabot center of gravity.

UYSABOT = Y-velocity of the sabot center of gravity.

UROTSABOT = Anti-clockwise velocity of the sabot.

UXPROJ = X-velocity of the projectile center of gravity.

UYPROJ = Y-velocity of the projectile center of gravity.

UROTPROJ = Anti-clockwise velocity of the projectile.

Output During Collision

IMPULSET = Impulse on the sabot in the tangential direction.

IMPULSEN = Impulse on the sabot in the normal direction.

IMPULSER = Angular impulse on the sabot.

VEL SLIP = Slip (separation) velocity in the tangential direction.

VEL APRO = Approach velocity in the normal direction.

VEL SEPA = Separation velocity in the normal direction.

CONTACTX = X-coordinate of the contact point.

CONTACTY = Y-coordinate of the contact point.

SABOT CORNER = Number of the sabot corner contacting the projectile.

Output During Sliding Contact

VELHN = Normal velocity of contact point H

FH = Normal contact force between sabot and projectile

ACX = Acceleration of mass center of sabot in x direction

ACY = Acceleration of mass center of sabot in y direction

ACT = angular acceleration of mass center of sabot in z direction
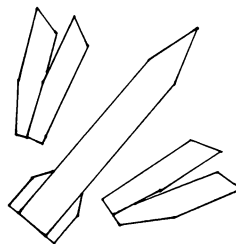
Figure 1.   Projectile and four sabot components.

initial   sliding   free   collision
contact   contact   flight
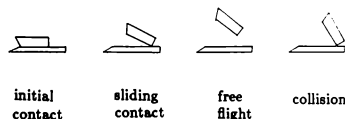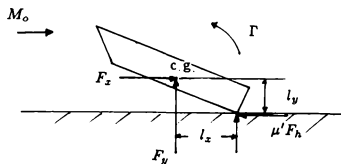
Figure 2.   Potential phases of sabot discard.

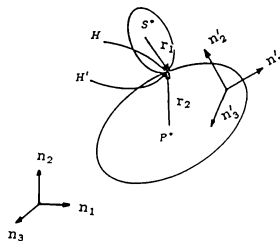Figure 3. Sketch of model of sliding contact.

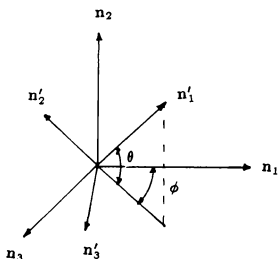Figure 4. Configuration at the time of collision.
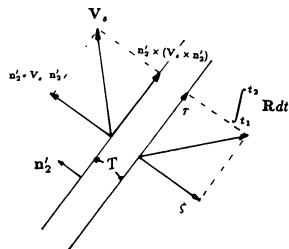

Figure 5. Coordinate transformation.

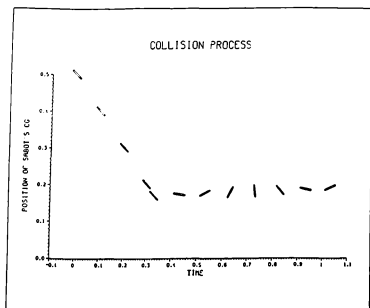Figure 6. Separation velocity and impulse referred to the collision plane.

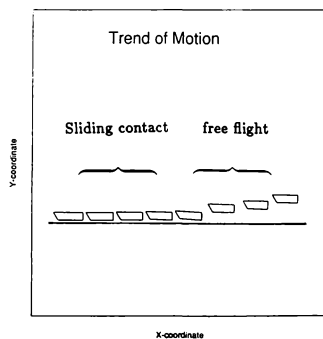

Figure 7. Collision process of case 1A.

Figure 8. Sliding contact and free flight of case 3.

Table 1. Summary of all cases

Case	Friction Coeff.	Coeff. of Rest.	Initial Sabot Position DX (m)	DY (m)	DROT (rad)	Initial Sabot Velocity UX (m/sec)	UY (m/sec)	UROT (rad/sec)	Number of collisions	Final Time (sec)	Final Sabot Velocity UX (m/sec)	UY (m/sec)	UROT (rad/sec)
1A	.30	.80	-.5	.5	-.7854	1.	-1.	0.	1	1.5	.692	2.65E-2	6.20
1B	.03	.80	-.5	.5	-.7854	1.	-1.	0.	2	1.5	.946	.591	-.925
1C	.30	.25	-.5	.5	-.7854	1.	-1.	0.	3	2.0	.676	5.19E-2	-.405
2A	.30	.80	0.	.5	1.E-4	0.	-1.	0.	3	1.5	-2.58E-2	.563	1.11
2B	.30	.80	0.	.5	0.	0.	-1.	0.	3	1.5	-2.58E-2	.563	1.11
2C	.30	.80	0.	.5	-1.E-4	0.	-1.	0.	3	1.5	2.58E-2	.563	-1.11
3	.30	—	-4.37E-2	5.15E-2	-1.0E-7	0.	0.	0.	Sliding Contact	8.5E-7	1.18E-2	1.02E-2	-4.45E-2

Table 2. Time history of collision for case 1A

```
INPUT DATA :
Q(L), L=1,6
-5.0000E-01  5.0000E-01 -7.8540E-01  0.0000E+00  0.0000E+00  0.0000E+00
U(L), L=1,6
1.0000E+00 -1.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
SM         PM         SJ         PJ         TOTIME      TIMEIV
4.5000E-01 4.5000E-01 4.8000E-03 5.0000E-01 1.5000E+00  1.0000E-01
R(L),L=1,4                                  COES        COER
1.8000E-01 1.8000E-01 1.8000E-01 1.8000E-01 3.0000E-01 3.0000E-01
THETA(L),L=1,4                              COER        RR
1.4399E-01 2.9976E+00 -2.9976E+00 -1.4399E-01 8.0000E-01 2.5000E-02

OUTPUT DATA:
TIME        DXSABOT    DYSABOT    DROTSABOT  DXPROJ     DYPROJ     DROTPROJ
GAP         UXSABOT    UYSABOT    UROTSABOT  UXPROJ     UYPROJ     UROTPROJ

0.0000E+00 -5.0000E-01  5.0000E-01 -7.8540E-01  0.000E+00  0.000E+00  0.000E+00
3.3077E-01  1.0000E+00 -1.0000E+00  0.0000E+00  0.000E+00  0.000E+00  0.000E+00

1.0000E-01 -4.0000E-01  4.0000E-01 -7.8540E-01  0.000E+00  0.000E+00  0.000E+00
2.3077E-01  1.0000E+00 -1.0000E+00  0.0000E+00  0.000E+00  0.000E+00  0.000E+00

2.0000E-01 -3.0000E-01  3.0000E-01 -7.8540E-01  0.000E+00  0.000E+00  0.000E+00
1.3077E-01  1.0000E+00 -1.0000E+00  0.0000E+00  0.000E+00  0.000E+00  0.000E+00

3.0000E-01 -2.0000E-01  2.0000E-01 -7.8540E-01  0.000E+00  0.000E+00  0.000E+00
3.0774E-02  1.0000E+00 -1.0000E+00  0.0000E+00  0.000E+00  0.000E+00  0.000E+00

3.3077E-01 -1.6923E-01  1.6923E-01 -7.8540E-01  0.000E+00  0.000E+00  0.000E+00
3.7812E-07  1.0000E+00 -1.0000E+00  0.0000E+00  0.000E+00  0.000E+00  0.000E+00

IMPACT :
IMPULSET =   -1.30578E-01 IMPULSEN =    4.61925E-01 IMPULSER =    2.97622E-02
VEL SLIP =    1.55677E+00 VEL APRO =   -1.00000E+00 VEL SEPA =   8.00000E-01
CONTACTX =   -6.15277E-02 CONTACTY =    2.50004E-02 SABOT CORNER =   4

3.3077E-01 -1.6923E-01  1.6923E-01 -7.8540E-01  0.000E+00  0.000E+00  0.000E+00
3.7812E-07  6.9205E-01  2.6500E-02  6.2005E+00  3.080E-02 -1.027E-01  4.991E-02

4.3077E-01 -1.0002E-01 -1.7188E-01 -1.6535E-01  3.080E-03 -1.027E-02  4.991E-03
1.0200E-01  6.9205E-01  2.6500E-02  6.2005E+00  3.080E-02 -1.027E-01  4.991E-02

5.3077E-01 -3.0816E-02  1.7453E-01 -4.5469E-01  6.159E-03 -2.053E-02  9.983E-02
7.0465E-02  6.9205E-01  2.6500E-02  6.2005E+00  3.080E-02 -1.027E-01  4.991E-02

6.3077E-01  3.8389E-02  1.7718E-01  1.0747E+00  9.239E-03 -3.080E-02  1.497E-02
1.4501E-02  6.9205E-01  2.6500E-02  6.2005E+00  3.080E-02 -1.027E-01  4.991E-02

7.3077E-01  1.0759E-01  1.7983E-01  1.6948E+00  1.232E-02 -4.106E-02  1.997E-02
1.4084E-02  6.9205E-01  2.6500E-02  6.2005E+00  3.080E-02 -1.027E-01  4.991E-02

8.3077E-01  1.7680E-01  1.8248E-01  2.3148E+00  1.540E-02 -5.133E-02  2.496E-02
5.3654E-02  6.9205E-01  2.6500E-02  6.2005E+00  3.080E-02 -1.027E-01  4.991E-02

9.3077E-02  2.4600E-01  1.8513E-01  2.9349E+00  1.848E-02 -6.159E-02  2.995E-02
1.4793E-01  6.9205E-01  2.6500E-02  6.2005E+00  3.080E-02 -1.027E-01  4.991E-02

1.0308E+00  3.1521E-01  1.8778E-01  3.5549E+00  2.156E-02 -7.186E-02  3.494E-02
1.3441E-01  6.9205E-01  2.6500E-02  6.2005E+00  3.080E-02 -1.027E-01  4.991E-02

1.1300E+00  3.8441E-01  1.9043E-01  4.1750E+00  2.464E-02 -8.212E-02  3.993E-02
6.9606E-02  6.9205E-01  2.6500E-02  6.2005E+00  3.080E-02 -1.027E-01  4.991E-02

1.2308E+00  4.5362E-01  1.9300E-01  4.7950E+00  2.772E-02 -9.239E-02  4.492E-02
6.2063E-02  6.9205E-01  2.6500E-02  6.2005E+00  3.080E-02 -1.027E-01  4.991E-02

1.3308E+00  5.2282E-01  1.9573E-01  5.4151E+00  3.080E-02 -1.027E-01  4.991E-02
9.1253E-02  6.9205E-01  2.6500E-02  6.2005E+00  3.080E-02 -1.027E-01  4.991E-02

1.4308E+00  5.9203E-01  1.9838E-01  6.0351E+00  3.387E-02 -1.129E-01  5.490E-02
1.7739E-01  6.9205E-01  2.6500E-02  6.2005E+00  3.080E-02 -1.027E-01  4.991E-02

1.5308E+00  6.6123E-01  2.0103E-01  6.6551E+00  3.695E-02 -1.232E-01  5.990E-02
1.8198E-01  6.9205E-01  2.6500E-02  6.2005E+00  3.080E-02 -1.027E-01  4.991E-02
```

Table 3. Time history of sliding contact and free flight for case 3

```
INPUT DATA
Q(L), L=1,6
-4.3742E-02  5.1524E-02 -1.0000E-07  0.0000E+00  0.0000E+00  0.0000E+00
U(L), L=1,6
0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
THO        SM         PM         SJ         PJ         TIMEIV
1.1900E+01 2.1970E-01 2.0000E+00 2.4988E-03 3.0000E-01 1.2076E-07
R(L),L=1,4                                  COES        COER
1.8300E-01 2.1300E-01 1.5326E-01 1.8400E-01 3.0000E-01 3.0000E-01
THETA(L),L=1,4                              COER        RR
1.3476E-01 3.0254E+00 -2.9704E+00 -1.4236E-01 8.0000E-01 2.5400E-02

OUTPUT DATA:
TIME            DXSABOT         DYSABOT         DROTSABOT
GAP             UXSABOT         UYSABOT         UROTSABOT
VELHN           ACX             ACY             ACT
FH              FX              FY              TOK

0.000000000E+00 -4.374200000E-02  5.152384000E-02 -1.000000000E-07
1.862139835E-05  0.000000000E+00  0.000000000E+00  0.000000000E+00
0.000000000E+00  9.059832142E+03  1.700879301E+04 -9.338566795E+04
5.510382274E+02  2.156000000E+03  3.185000000E+03 -3.920000000E+02

1.207564800E-07 -4.374199993E-02  5.152384012E-02 -1.006808609E-07
1.862139835E-05  1.094033439E-03  2.053921973E-03 -1.127692454E-02
2.004420935E-13  1.169491472E+04  1.528783354E+04 -8.393685229E+04
4.107427834E+02  2.695000000E+03 -3.940000000E+03 -3.185000000E+02

2.415123600E-07 -4.374199972E-02  5.152384048E-02 -1.026546315E-07
1.862139835E-05  2.506270174E-03  3.900026939E-03 -2.141284337E-02
1.303742270E-12  1.302989270E+04  1.343993556E+04 -7.379108904E+04
2.577589011E+02  2.940000000E+03  2.695000000E+03 -2.450000000E+02

3.622694400E-07 -4.374199932E-02  5.152384105E-02 -1.057703866E-07
1.862139835E-05  4.079714150E-03  5.522986248E-03 -3.032359554E-02
3.776519895E-12  1.436487067E+04  1.159203759E+04 -6.364532587E+04
6.077502145E+01  3.185000000E+03  2.450000000E+03 -1.715000000E+02

4.830259200E-07 -4.374199872E-02  5.152384180E-02 -1.099041994E-07
1.862139835E-05  5.814365368E-03  6.922799903E-03 -3.800918106E-02
7.870309439E-12  1.561217170E+04  1.003639610E+04 -3.921822929E+04
0.000000000E+00  3.430000000E+03  2.205000000E+03 -9.800000000E+01

6.037824000E-07 -4.374199791E-02  5.152384271E-02 -1.147799969E-07
1.862141945E-05  7.699636269E-03  8.134759768E-03 -4.274503638E-02
3.493951549E-04  1.672732683E+04  8.921240974E+03 -9.804557324E+03
0.000000000E+00  3.675000000E+03  1.960000000E+03 -2.450000000E+01

7.245388800E-07 -4.374199685E-02  5.152384375E-02 -1.200132227E-07
1.862151366E-05  9.719569376E-03  9.212057425E-03 -4.392900023E-02
1.211051638E-03  1.784248195E+04  7.806085852E+03 -4.902278662E+03
0.000000000E+00  3.920000000E+03  1.715000000E+03 -1.225000000E+01

8.452953600E-07 -4.374199555E-02  5.152384493E-02 -1.253536770E-07
1.862171031E-05  1.187414469E-02  1.015469288E-02 -4.452039213E-02
2.045664504E-03  1.895763707E+04  6.690930731E+03 -9.804557324E+03
0.000000000E+00  4.165000000E+03  1.470000000E+03 -2.450000000E+00
```

231

UTILIZING A BLADE ELEMENT MODEL FOR HELICOPTER PILOT TRAINING

Martin T. Jakub, Leonard Richmond, Allen Tracy
Eyring, Inc.
Salt Lake City, Utah

ABSTRACT

Advances in computer technology have allowed blade element engineering simulations to be done in real-time, making their use in pilot training a possibility. This paper describes how the modelling fidelity can be increased to provide better training for pilots in disturbed air environments near ships, other aircraft, or the ground, and in power settling conditions. Historically, the shipboard operations environment has been difficult to simulate because of a lack of both experimental data and the computing power necessary to simulate fluid dynamics in real-time. Our approach utilizes analytic functions to model known existing flow patterns. We show a method of describing the velocity field in the vicinity of the rotors, suitable for communication to dedicated blade element computer systems.

INTRODUCTION

In the mid to late '60s, research was done into carrier airwakes, eventually resulting in the carrier landing disturbance model described in MIL-F-8785C and MIL-STD-1797. This research was designed to understand airflows that were complicating fixed-wing aircraft carrier landing, particularly the burble or "rooster tail" that occurs just before the plane crosses the deck. Eventually the pilot-training simulators for fixed-wing aircraft incorporated models based on the studies and/or the mil standards[1,2]. For rotary aircraft, however, the major piloting difficulties result from wind patterns caused by the ship superstructure. These difficulties are further complicated by the fact that helicopters land on many types of ships, not just carriers.

The safety envelope for shipboard helicopter operations has been explored by Naval Air Test Center. Given a specific ship/aircraft combination, the ability of the pilot to perform various tasks is evaluated. The pilot faces the cumulative obstacles of ship motion, ship airwake turbulence, obstructions to field of view or landing aids, and wind over deck factors. Such evaluations, called "dynamic interface studies," are expensive, and there are many ship/aircraft combinations to be evaluated. Could computer simulation be a cost-effective substitute for field tests? Unfortunately, there is not enough quantitative experimental data with which to develop models. If models could be developed and tested against adequate experimental data, such models could be used to estimate new ship/aircraft combinations. That time is not yet here. Presently, discussion continues among the experts in the simulation and fleet operations communities toward that end[3,4,5].

During the last 30 years, remarkable progress in the computer industry has made it possible to integrate the lift and drag along a rotary wing in real-time. Such a model is called a blade element model. Until now, such models have been used only in non-real time applications to design rotary aircraft and to design rotor maps (lookup tables) to be used in real-time flight simulation. Now it is possible to put the blade element model directly in the real-time simulation.

Naval Training Systems Center has tasked Eyring, Inc. with upgrading the pilot-training simulators for the CH46 tandem rotor helicopters, and specifically to utilize a blade element model and improve the simulation of the localized wind near ships and formation aircraft. The project is an effort by the government to improve simulation in light of the known difficulties of the task[6].

BLADE ELEMENT MODEL

To provide needed computing resources to add blade element modeling to the existing simulator, we have added a "compute box" dedicated to that task. It is connected to the main simulation computer via a parallel interface. All other simulation tasks (including fuselage aerodynamics) remain in the main simulation computer.

Using a separate compute box introduces several new concerns. A transport delay is one inevitable result, since data must be passed to the box on one iteration

and returned on the following iteration. Our main simulation runs at a relatively high iteration rate of 64 Hz, so the additional delay is only 1/64th of a second. Another concern is that of frequency folding. Since the main simulation effectively "samples" the output of the compute box, it is possible that high frequency components of the aeroforces could be folded into lower frequencies. When this difficulty was studied at NASA Ames, it was found that these problems can be largely alleviated by running the blade element model at an even higher frequency, and by using filters judiciously[7]. The blade element model runs at three times 64 Hz, which prevents aliasing of all harmonics through the 7th of the 6-per-revolution beat experienced by the tandem 3-blade rotor hubs of the CH46, thus minimizing the problem.

The performance specifications for the blade element model are for an azimuth step of 8.25 degrees and 10 elements per blade, 3 blades per hub, and 2 hubs per helicopter. To meet this requirement, the compute box is designed using a VME bus chassis with digital signal processor (DSP) cards. DSPs are a fairly recent development in chip technology, specifically designed to perform signal processing. What makes DSPs ideal in a compute box for integration is that a single chip can perform a floating point addition and multiplication concurrently in a single cycle. The compute box contains a total of six DSPs. Maximum peak performance is 192 million floating point operations per second.

The inputs to the blade element model are passed to the main DSP for each of the two hubs via a 68030 processor, which handles communications between the individual DSPs and between the compute box and the main simulation computer. Calculations general to a hub are completed and results are passed to subservient DSPs, which then perform the calculations for the other blades. These results are passed back to the main DSPs to be summed. The aft hub calculations are passed to the forward hub's main DSP to be included in the total force and moment results and passed back to the host via the 68030. Since most of the time is spent performing the individual blade calculations, the six DSPs form a very efficient parallel computer system for blade element calculations.

The software for the blade element model is written in ADA and is based on the GenHel blade element program[8]. The GenHel program was developed by Sikorsky in the 1970's and has been actively used by NASA Ames for studies of rotorcraft[9]. In short, the GenHel model has been evolving and improving for a

number of years, and constitutes a solid starting point for further development. The software model was converted from Fortran to ADA, modified for the rotor hub geometry of the CH46, and enhanced to account for dynamic stall and perturbed airflow effects. Static tests from wind tunnels do not accurately predict the stall conditions or even the lift under non-stall conditions for a wing that is cyclicly changing in pitch, since it takes a certain amount of time for the flow to become the more turbulent flow associated with a stall condition or to settle to the laminar flow associated with a static condition. The stall effect is referred to as "lift overshoot" and is important for simulating flight that is at the edge of the speed envelope[10], whereas the non-stall lift correction is necessary even for normal flight. Other modifications were necessary to allow thrust and drag gains for power settling and ground effect conditions, and to simulate local wind velocities near the hub.

THE WIND TO BLADE ELEMENT INTERFACE

The parallel interface to the blade element compute box is of limited bandwidth, so a short and succinct list of input and output variables is a necessity. To describe the localized winds near a ship, the entire wind velocity field must be passed. One way to do this is by approximating the velocity field in a linear fashion; i.e., use the first order multi-dimensional Taylor series approximation:

$$V(x+dx) = V(x) + V'(x)*dx \qquad (1)$$

where:

$V = (u,v,w)$ is a vector function of position, and
$V' = $ Jacobian matrix defined by
$$\nabla u$$
$$\nabla v$$
$$\nabla w$$
i.e., the matrix whose rows are the gradients of the components of V.

Although the first order approximation is very simple, it allows for surprisingly complex flow fields. For example, the Jacobian matrix:

$$\begin{array}{ccc} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{array}$$

describes a circular flow around the x axis. A circular or elliptical flow is especially applicable to modelling the standing and trailing vortices that can occur around

the ship superstructure, especially near a hangar.

The main simulation computer calculates the Jacobian matrix at each rotor hub and the velocity at the rotor hubs. From this information, the blade element model processors can estimate the velocity field at any region of the rotor disk, using equation (1). The Jacobian itself is estimated by putting an imaginary "pizza box" around the nominal region of the rotor disk, and then evaluating the wind velocities at the centers of the faces of the box. The differences in velocities on opposite faces, divided by the distance between the faces, gives an estimate of the partial derivatives (one column in the Jacobian matrix). For the two Jacobians and hub velocities, a total of 20 floating point words describes the wind velocity fields in the rotor regions. This is a small enough interface to meet the bandwidth requirements.

The Jacobian matrix is used to calculate the localized wind effects on the fuselage. The yaw component of the aerodynamic torque on the fuselage is

$$N = q^*C_N - K^*(r + (\nabla \times V)_z) \qquad (2)$$

where:

q = dynamic air pressure
C_N = yaw coefficient from wind tunnel data
K = damping factor
r = turn rate
$(\nabla \times V)_z$ =z component of the curl of the wind velocity

The curl of a vector field is a linear sum of the non-diagonal elements of the Jacobian, and intuitively represents the rotational component of the flow. The K*r term represents the damping that occurs when the fuselage is rotating in wind with no rotational components. Similarly, a nonrotating fuselage with the wind rotating about the aircraft will also result in a torque. Consider an example situation where this would arise. Imagine taking off from a ship that is facing into the wind. The helo pad is directly downwind of a hangar that acts as a wind shield. If the helo is hovered to a position where the nose is exposed to the wind but the aft section is still in the wind shadow, a torque will occur and the helo will turn.

WINDS

Modelling of the ship-wind interface can be divided into two basic areas: turbulence and the ship-airwake structure[17].

Turbulence

There are two widely accepted statistical models for free-air turbulence: the Dryden and Von Karman models. These models construct an energy spectrum based upon the correlations of spatial wavelengths. In other words, the energy of a particular vector in space is affected by those vectors around it[14].

A vector may be influenced by vectors in all three dimensions (x,y,z). However, the effects of cross-spectra, such as the influence of a vector in the z direction upon one in the x direction, are not noticeable to pilots[19]. Therefore, only auto-spectra are considered.

The Von Karman forms are considered to be more accurate than the Dryden forms. The Von Karman equation for the longitudinal auto-spectrum (same direction as the wind) is

$$\Phi(\Omega) = \frac{\sigma^2 L}{\pi} \times \frac{1}{[1 + (aL\Omega)^2]^n} \qquad (3)$$

where:

n = 5/6
σ = the standard deviation or turbulence intensity
L = turbulent length scale
Ω = the spatial frequency
a = 1.339

The lateral and vertical auto-spectra equations are similar to the equations above.

These spectrum equations describe atmospheric turbulence, but do not represent the interaction of turbulence with a ship at sea. Due to the low altitude and interaction with the ship, the equations will be altered and must generate velocities from random frequencies.

Dr. Val Healy, of the Naval Postgraduate School, has revised the Von Karman equations to include these considerations[16]:

$$\Phi(\Omega) = \frac{4\sigma^2}{(1 + 70.8\bar{n}^2)^n} \qquad (4)$$

where

$\bar{n} = \Omega L/U$
U = wind velocity
n = 5/6

A digital filter is chosen with a transfer function whose magnitude squared matches this spectrum. This filter is then applied to white noise to yield a simulated turbulence with the appropriate energy spectrum[19].

Modelling Winds Near a Superstructure

The wind whipping over the deck also produces a steady-state airflow pattern behind the superstructure of the ship. This steady state flow consists of various vortices and a component above the deck in the direction of the wind.

Becuase this flow pattern is very complex and an adequate experimental data is currently unavailable, an accurate statistical model has been impossible to devise[6].

A previous attempt at a ship-airwake model was made by Fortenbough using a method called Strouhal scaling. Data was gathered from a Boeing wind tunnel experiment on a 1/50 scale model of an FF 1052 class destroyer. The data was put into table form and then scaled to fit other ships[18]. This approach has two problems. First, it assumes that deck structures of ships are similar, which in most instances is not the case. Second, wind speeds in the tunnel were measured at points too far apart to get an accurate idea of flow pattern, especially that of vortices. Healy states that the method is very crude and that "results can be expected to be as accurate as picking random numbers"[16].

An alternate method is to pattern the airflow around the ship superstructure from the study of aerodynamics of buildings. Structures or groups of structures on ship decks can be characterized as bluff bodies, which are described as short, squat buildings. From this and studies of helium bubble patterns behind models in wind tunnel tests a general description of the airflow can be devised[15].

With wind heading at 0° relative to the ship, a standing vortex forms directly behind the hangar (figure 1). A cross-section of this vortex is approximately circular with a diameter equal to the height of the hangar. This vortex extends clear across the face of the hangar. There are also two trailing vortices which start along the sides of the hangar and flow back along the side of the ship until they die out. Behind the standing vortex and between the trailing vortices, the pattern dives down towards the deck, flows back, and rises a distance aft of the ship.

When the wind moves off the 0° heading, this vortex pattern is somewhat altered. For instance, if the wind blows from 330°, as in figure 2, the standing vortex behind the hangar will begin to shorten on the left side.
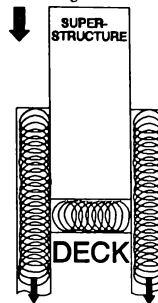
Figure 1. Wind from 0° - Vortices
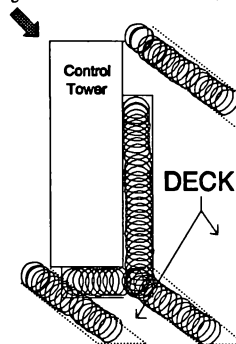
Figure 2. Wind from 330° - Vortices

At the same time a standing vortex forms on the right side of the hangar, beginning at the aft corner and growing toward the front corner as the wind angle decreases towards 270°. The trailing vortices now flow downwind from the front right and aft left corners of the hangar. In addition, a third trailing vortex forms where the two standing vortices meet at the right aft corner. This vortex is weaker than the other two. The "deck" component of the airflow shows the same general behavior as for 0° heading wind as it flows downwind. The vortices line up with the relative wind as its angle changes.

An analytical model of the ship airwake has been designed, based upon this flow pattern and the previously mentioned Boeing data[13]. The Boeing data is used as "starting points." In other words, with a mean wind speed of 20 feet/second, the data may show an x-component velocity of 6 feet/second at a point on the perimeter of the standing vortex. This is used as the starting velocity, and the conservation of momentum is used to calculate x-component velocity as we move further inside the vortex. Again, if a certain velocity is measured at a point just aft of the vortex, it is used as a starting point and is then manipulated mathematically in accordance with the flow pattern.

This analytical model is dependent on the factors of mean wind speed, wind heading in relation to ship heading, and the dimensions of the ship and its superstructure. Therefore, it can be applied to any ship by changing only those inputs contingent on the dimensions of the specific ships.

Combining the Wind Elements

As mentioned previously, to estimate the Jacobian matrix of the wind velocity at the hubs, it is necessary to calculate the wind velocities in the rotor sampling box shown in figure 3.
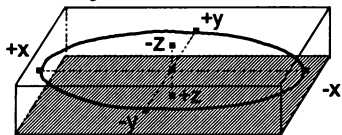
Figure 3. Rotor Sampling "Box"

For each point, a gain value is calculated to indicate if it is in the region of a particular wind component (eight vortices and the deck component). If a value greater than 0 is computed, velocities are then calculated for that wind component.

Three arrays of size 3x14 (x,y,z component velocities for each point) are used to store velocities for each of the three wind component types (standing vortices, trailing vortices, and deck component). These are then added together along with turbulence velocities to form one 3x14 array.

Formation Aircraft Disturbance

This module simulates the disturbances encountered by a helicopter flying in formation behind another helicopter. The formation aircraft, which is also a CH46 tandem rotor helicopter, is modeled as a disc whose radius is the distance from the center of gravity to the rotor tip. Flight instructors have determined that the disturbance envelope may be represented as a cylinder (figure 4) which trails behind and below the lead aircraft at an angle determined by the following equation:

$$\gamma = atan(w/u) \qquad (5)$$

where:
w = vertical downwash velocity
u = forward velocity of the aircraft

γ = skew angle
Ψ = formation aircraft heading
Vd = direction of downwash
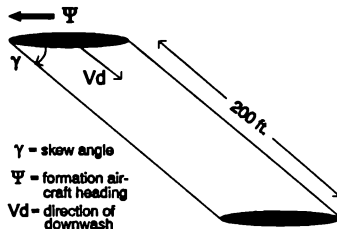
Figure 4. Formation Aircraft Disturbance Envelope

Pilots have stated that the wash tends to roll trailing aircraft in towards the center of this cylinder as in figure 5.

γ = skew angle
Ψ = formation aircraft heading
Vd = direction of downwash
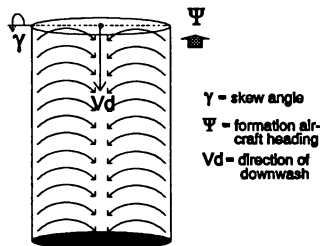
Figure 5. Formation Aircraft Disturbance Pattern

This effect is because of wing-tip vortices shed from the lead aircraft. This effect is to be simulated by constructing two vortices, one on either side of the radial axis. The magnitude of the vortex velocities will dissipate down the cylinder length and will completely disappear at a distance at a distance of 200 feet. In addition, the wash produces a strong wind component in the direction of the cylinder axis which dissipates at 200 feet down the axis and radially away from the

center. Thus, rolling affects will become more severe towards the center of the cylinder. All affects are washed to mean wind values at a distance of 2 cylinder radii from the center.

The location of each point is computed in the coordinate system of the trailing cylinder. The cylinder is then tilted to form a right cylinder to more easily facilitate construction of the vortices. Velocities are initially calculated in vortex coordinates and are then converted back to cylinder coordinates and finally to earth coordinates.

In addition to these steady state affects, three-dimensional turbulence is added in proportion to the component velocities already calculated.

Two velocity arrays and two Jacobian matrices are computed in the same way as those in the ship airwake routine.

Ground Effects

When the helicopter is within 50 to 75 feet of the ground, ground effects from the rotor downwash can significantly decrease required torque. Existing models calculate "in hover" ground effects, ignoring the fact that lateral movement decreases ground effects.

A "mirror" model has been devised to address this problem (figure 6). Each rotor is modeled as a separate disc. Downwash velocities are passed from the BEM and a cylinder is constructed for each disc in much the same way as in the Formation Aircraft Module. The cylinders extend to the ground and are reflected back up. If the aircraft is in hover, the cylinders reflect straight up, and will result in full ground effects.

Ψ = trainer heading
γ = skew angle
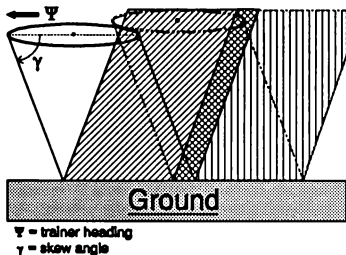
Figure 6. Ground Effect Mirror Model

If there is a forward velocity, the cylinders will reflect backwards resulting in less effect on the forward hub. The aft hub will experience increased effects from the forward cylinder and decreased effects from the aft cylinder.

The cylinders are analyzed to check the amount of area which intersects with the rotor discs. Four gains are calculated based upon the amount of intersecting area. These gains are multiplied by the hover thrust gains to produce a total thrust gain:

$$GETHR(1) = (GHW(1)+GHW(2)) * (GTH(1)-1) + 1$$
$$GETHR(2) = (GHW(3)+GHW(4)) * (GTH(2)-1) + 1$$

(6)

where:

GETHR = the total thrust gain of the hubs
GHW = gains from amount of area in discs
GTH = in hover thrust gains

Calculating the Jacobians and Velocities

Finally, the 3x14 arrays from both the Ship Airwake module and Formation Aircraft Disturbance module are added together. The sample points at the hub locations are put into two velocity arrays, one for the forward hub and one for the aft hub. Component velocities at the remaining points are used to form the Jacobian matrices for each hub. The two velocity arrays and two Jacobian matrices are passed to the Blade Element Model for application.

CONCLUSIONS AND FUTURE DEVELOPMENTS

The project described in this paper is currently being implemented. Soon pilots and instructors will be able to provide their opinions as to the accuracy and value of the ship operations simulation for windy conditions. There is also considerable interest as to how the fidelity of a blade-element based simulation will compare to the rotor-map model presently employed. We have high hopes that a blade-element model will provide a better global model of flight, especially for the edge-of-the-envelope conditions that are not considered in the design of rotor maps.

For the immediate future, localized wind will be simulated by using functions. Table-based lookups are memory-intensive and rely on vast amounts of data collected for each specific ship. The solution of fluid dynamics equations in real-time is at present science fiction and is not a viable alternative. At some future

point, this fiction will become fact. At that time, it will make sense to use the visual data base to solve for wind flow patterns that match the constraints of the terrain and objects in the database.

Meanwhile, we can hope that enough quantitative data will be collected to provide a better understanding of airflow around ships. With such data at hand, the task will be to improve the simulation until it can be used to predict or extrapolate the difficulty of shipboard operations by those doing dynamic interface studies.

REFERENCES

1. Weir, D.H., "Analysis and Comparison of Carrier Airwakes," Systems Technology, Inc., STI Technical Report No. 158-1, April 1966.

2. Durand, T.S., "Carrier Landing Analysis," Systems Technology, Inc., STI Technical Report No. 137-2, February 1967.

3. Carico, D., McCallum, K.E., Higman, J., "Dynamic Interface Flight Test and Simulation Limitations," Eleventh European Rotorcraft Forum, Paper No. 100, London, England, September 1985.

4. Anon., "Helicopter/Ship Dynamic Interface (DI) Working Group Meeting Proceedings," Naval Air Test Center, April 1988.

5. Anon., "Minutes of the First Aerodynamic Ship Interface Conference," Washington D.C., November 1989.

6. Galloway, R.T., Frey, F.E., Carrico, D., "The Ship Environment Simulation Problem," AIAA paper No. 3171-CP, September 1990.

7. McFarland, R.E., "The N/Rev Phenomenon in Simulating a Blade-Element Rotor System," NASA TM-84344.

8. Howlett, J.J., "UH-60A Black Hawk Engineering Simulation Program," NASA CR-166309, December 1981.

9. Houch, J.A., Moore, F.L, Howlett, J.J., Polecat, K.S., and Browne, M.M., "Rotor System Research Aircraft Simulation Mathematical Model," NASA TM-87629, 1977.

10. Prouty, R.W., Helicopter Performance, Stability,

and Control, pp. 220-225, PWS Engineering, 1986.

11. Campbell, Warren C. "A Spatial Model of Wind Shear and Turbulence for Flight Simulation," NASA Technical Paper No. 2313, May 1984.

12. Cermak, J., "Aerodynamics of Buildings," Annual Review of Fluid Mechanics, Vol. 8 (1976), pp. 75-106.

13. Garnett, T.S., "Investigation to Study Aerodynamic Ship Wake Turbulence Generated by an FF1052 Frigate," Boeing Vertol Report D210-11140-1, Dec. 1976.

14. Handbook of Turbulence, Volume 1. Frost, Walter and Moulden, Trevor H., editors. Plenum Press. NY.

15. Healy, J. Val, Ph.D., & Johns, LCDR, Michael, "The Airwake of a DD-963 Class Destroyer", Naval Engineers Journal, May 1989, pp. 36-42.

16. Healy J. Val. "The Prospects for Simulating the Helicopter/Ship Interface", Naval Engineers Journal, March 1987, pp. 45-60.

17. Military Specification. "Flying Qualities of Piloted Airplanes," MIL-F-8785C, Nov. 5, 1980.

18. Fortenbaugh, Robert A., "Progress in Mathematical Modeling of the Aircraft Operational Environment of DD 963 Class Ships," AIAA Atmospheric Flight Mechanics Conference, Boulder, CO, Aug. 6-8, 1979.

19. Robinson, Paul A., "Modelling Atmospheric Effects for Flight Simulators," AIAA flight simulations conference paper 89-3284-CP, August 1989.

COMPUTATIONAL REQUIREMENTS AND MODEL REFINEMENT CONSIDERATIONS FOR MODERN BLADE-ELEMENT ROTOR MODELS WITHIN HELICOPTER FLIGHT TRAINING SIMULATORS

Stanley J. Briczinski, Staff Scientist
CAE-Link Corporation
Binghamton, New York

Abstract

Under an internal research and development initiative, CAE-Link has recently supported work aimed at the development of a blade-element rotor model for application in helicopter training simulators. Part of this effort included a sensitivity analysis to determine allowable simplifications of this new rotor model without sacrificing discernible accuracy of the resultant flying qualities characteristics of the simulator. The work of J.A. Houck[1] was studied, and effort to update the findings of this previous work was undertaken. Emphasis was placed on the requirements of a training simulator rather than an engineering simulator when conducting the new sensitivity analysis. Criteria were developed for evaluating the performance of the helicopter simulation under these rules. Basically, Houck's findings were reconfirmed, and some additional discoveries and refinements were made.

The results of the new sensitivity analysis suggest that the generalization of conclusions regarding blade-element rotor model requirements should not be sought; model optimization is dependent upon too many variables, as well as characteristics of the particular helicopter and rotor system itself. A better approach is recommended whereby the simulator manufacturer creates a standing tool which provides easy and immediate access at the beginning of a helicopter simulator design project to test desired model simplifications (compared to an established benchmark obtained for the modeling case with little or no simplification) for the particular helicopter simulation application at hand. Such a standing design tool should embrace necessary frequency domain tools as well as traditional time domain analysis techniques, to assist in the model simplification analysis task. Particular care must be taken to address the problem of aliasing of high-frequency rotor characteristics into the lower frequency domain of pilot handling qualities.

Introduction: Purpose of Study

In recent years, customer specifications for helicopter training simulators have become increasingly demanding. To meet this demand, simulator man-ufacturers are improving many aspects of their products. In the case of the aerodynamic math model, adoption of a blade-element method of modeling rotor characteristics promises higher accuracy and appears very attractive to users. Although this type of model has been known to the helicopter industry for some time, until recently it has not been well-suited for real-time simulator applications. Now, new computer resources and program-hosting techniques make real-time solution of this computationally demanding rotor model a practical possibility.

The state-of-the-art blade-element rotor model offers more potential than any other rotor modeling technique for improving kinematic and dynamic helicopter flight characteristics within a simulator. The sophisticated blade-element rotor model is particularly adept as a helicopter design tool, and as such has been used primarily in the context of helicopter engineering-type simulators. In recent years, this rotor modeling technique has been introduced into the realm of helicopter training simulators. Characteristics of training simulators and engineering simulators are becoming more alike, as driven by increasingly demanding requirements for these two different applications. For example, the need to better model pilot workload within the engineering simulator during control system design has made the engineering simulator more resemble the cockpit detail of training simulators, while the need to better replicate true aircraft flying qualities within the training simulator have made the flight model of the training device more closely resemble the detail found in engineering simulators.

Yet differences remain in the requirements of training simulators, as compared to engineering simulators. The rotor model of a helicopter training simulator should be as concise as possible for a number of reasons. The model should be short to enhance real-time execution, to require low-cost computational facilities, and to reduce software complexity in order to simplify model analysis and refinement tasks, while improving the reliability of the training device. Reducing computational requirements is particularly important to reduce procurement costs if multiple units of a specific simulator are being purchased. Thus it is desirable to optimize a blade-element rotor

model, while simultaneously maintaining the accuracy offered by this modeling method. A sensitivity analysis to determine acceptable simplifications which could be made to the blade-element rotor model would accomplish this goal.

Houck[1] describes an earlier effort regarding a sensitivity analysis for real-time helicopter simulations. While the work of this reference is detailed and accurate, and centers on the rotor model itself, it is desirable to now update such a sensitivity analysis for two reasons. First, we need to concentrate on more practical values of rotor parameters subjected to the sensitivity analysis, according to today's trends (note that Houck's paper was written in 1976). Second, we need to construct acceptance criteria for evaluating tested simplifications which concentrate on the requirements of training simulators (i.e., some of Houck's particular analysis methodology is pertinent only to engineering simulators).

Background: The Blade-Element Model

The blade-element rotor model treats the motion of each blade of the helicopter rotor system separately and rigorously. Each blade is divided into a number of elements or segments. At each segment, the kinematical characteristics of the segment are used in conjunction with the local relative velocity flow field (which includes contributions from the rotor's own downwash and blade motion, as well as free stream velocity components) to determine the resultant aerodynamic forces acting on the segment. (Segment aerodynamic force characteristics are governed by appropriate airfoil lift and drag characteristics carried within the rotor model via traditional airfoil data tables.) For each blade the forces acting on the segments are summed and resolved to determine the net forces acting on the blade, which includes mass and inertia contributions as well as the aerodynamic contribution. The resultant shear reactions at the hub for each blade are calculated and summed to represent the net force and moment transferred from the rotor system to the rotor shaft. Meanwhile the total forces acting on each blade are used to determine the acceleration of the blade, which in turn is used to determine updated values of blade velocity and position. Control system inputs are sampled to establish the pitch of each blade.

There are a number of blade-element rotor models now in use within the industry, all of which adhere, in general, to the solution approach just described. After conducting a literature survey and studying a number of these rotor models, Link elected to independently derive the governing rotor blade equations

of motion and the equations dictating the inertial shears transferred from the blades to the rotor head. This was done to discipline and familiarize ourselves with the fundamental principles which govern the blade-element rotor model approach. It also afforded us the opportunity to examine simplications sometimes carried in other blade-element models, and to check for possible errors that may have been overlooked in previous derivations.

The model structure (or formalism) as well as the details of the particular aerodynamic force and moment calculations used in the Link blade-element rotor model were done to resemble the methodology of the Howlett blade-element rotor model.[2] This particular approach for implementing a discrete blade-element rotor model into a helicopter simulator was chosen based on our familiarity with this model, and its somewhat universal acceptance and recognition in the industry as a good rotor model for both engineering and training simulators.

The blade-element rotor model subjected to the sensitivity analysis and optimization process for training simulator applications in this study is that of an articulated rotor system, with fully rigorous flapping and lead-lag degrees of freedom. Lead-lag damper characteristics are modeled. Rigid blades are assumed within this rotor model.

Analysis Methods, Tools, and Considerations

The blade-element rotor model under investigation and optimization was incorporated into a complete helicopter flight simulation using the Universal Helicopter Model (UHM). The UHM is a Link design tool for developing and analyzing helicopter aerodynamic mathematical models. The UHM employs a modular or component formalism to describe the key systems of the helicopter aero/flight model (e.g., main rotor, control system, horizontal tail), allowing each of these individual components to be easily changed to represent variations in their general characteristics, or specific differences to represent details of particular helicopters. The UHM resides within the Link off-line software development computer facility. This off-line facility was used to conduct all studies of the blade-element rotor model, emulating a real-time operating environment, as needed, for the simulation.

Houck's sensitivity analysis study[1] utilized four types of tests:

1. Vehicle static trim comparisons

2. Total rotor force and moment comparisons

3. Blade parameter variation with azimuth

4. Dynamic response comparisons for the total vehicle

While this analysis approach is excellent, for training simulator requirements it is not necessary to conduct test types 2 and 3, or at least to dwell upon the results of such rotor-specific tests. We do not wish to draw conclusions regarding rotor model optimization for a training simulator based upon performance of the model in areas which may be transparent to the simulator user. Remember that training simulator usage is different from engineering simulator usage. Training simulators are not used to specifically analyze particular rotor characteristics, and need model the rotor only to the degree of accuracy necessary to produce proper and acceptable fuselage or net vehicle response. It is the vehicle response that the student pilot (or the instructor) reacts to and evaluates in order to judge the fidelity of the training simulator.

For a simulator to be an effective man-in-the-loop training device, its math model must accurately reproduce frequency response characteristics of the actual vehicle for characteristic frequencies approaching approximately 3 Hz. But the capability of the blade-element rotor model to calculate high-frequency rotor characteristics, combined with the computational restraints introduced by the discrete (rather than continuous) nature of the rotor math model solution within the context of a digital computer simulation, introduces the problem of aliasing. Depending upon the duty cycle or computational rate chosen for the simulation, non-negligible rotor force/moment characteristics of higher frequency can be aliased or folded over, appearing with incorrect lower frequency characteristics within the net simulation. This phenomenon is described in detail by McFarland.[3] When optimizing any blade-element rotor model for eventual real-time application, the aliasing problem must be addressed.

Three steps are recommended for addressing the aliasing problem:

1. Identify what characteristic frequencies of rotor forces will be aliased in the real-time digital simulation. (A formula for providing this information is available.[3] The aliased frequencies are a function of the rotor speed and computational rate of the real-time solution.)

2. Identify the relative magnitude of the aliased rotor force components in order to determine if they are significant enough to warrant correction in the real-time helicopter simulator

3. If the aliased rotor force components are large enough to contaminate the real-time solution in the pilot's frequency domain, apply techniques to correct for the aliased forces (again, methods for correcting such aliased forces have been devised[3]).

While we may wish to address only net vehicle trim and time history transient response when evaluating the rotor model within the context of training simulator requirements, the important aliasing problem is most efficiently addressed by studying the characteristics of the rotor within the simulation. Thus a two-pronged approach for studying the applicability of rotor models within the training simulator environment was established. First, rotor characteristics were studied for helicopter trim and dynamic response test cases, to determine that proper frequency content was being demonstrated, and to investigate any possible aliasing occurrences. After demonstrating acceptable rotor parameter frequency characteristics, the second analysis step was conducted: the net vehicle trim and response characteristics were evaluated to establish if changes in the rotor model parameters being investigated impacted the behavior of the helicopter simulation in a perceptible and non-negligible manner. The rotor parameter analysis step was itself conducted in two parts: analyses in the time domain and in the frequency domain. The net vehicle parameter analysis was conducted only in the time domain.

Examination of Rotor Power Spectrum

To analyze the frequency content of rotor parameters solved by the blade-element model, a Fast Fourier Transform (FFT) routine was built into the off-line blade-element development program. Parameters of interest were then passed through this FFT, which decomposed the parameters into components of sinusoidal signals and identified the characteristic frequency of each such sinusoid which composed the parameter. The power spectrum of each parameter under investigation revealed the frequency content regarding potential aliasing, and also indicated the impact that variation in a rotor model sensitivity analysis parameter had on the power content of these rotor parameters. Blade flapping and lead-lag angles and blade shear forces at the hinge (i.e., rotor forces ultimately transferred through the rotor hub to the fuselage) were the rotor parameters subjected to this analysis.

Portions of the spectral analysis of rotor parameters were conducted in both the rotor rotating and nonrotating frames of reference. Forces generated by a blade within the rotating rotor reference frame

were studied to decide if they were composed of components of proper frequency. Then the forces in the nonrotating (i.e., hub or fuselage) reference frame, resulting from the transfer of the sum of each force on each blade in the rotating frame across the attachment mechanism of the blade to the hub, were studied to determine if these nonrotating forces were composed of proper components according to the frequency content of the forces established in the rotating frame. This transformation to the nonrotating frame involves trigonometric functions, dependent on the number of blades of the rotor system.

In studying the frequency content of rotor forces in the rotating frame, two approaches were considered. One method was to simply obtain a trimmed helicopter solution and investigate the frequency characteristics of the blade forces as calculated by the baseline blade-element rotor model. In this condition, periodic variation in local airstream (in forward flight) about the rotor azimuth would produce blade forces which alternate at integer multiples of rotor speed. In hover, such periodicity disappears due to a constant aero environment at any rotor azimuth position. The proper frequency content characteristics were observed in the rotor model. The second method was to excite the control inputs (collective or cyclic) at some frequency (not necessarily integer multiples of rotor speed) and then analyze the content of the individual blade forces predicted by the blade-element rotor model to determine if proper frequency content was observed. This method also was pursued in hover and at forward speed. Results from the baseline blade-element rotor model were proper with respect to frequency content. Also, magnitudes were estimated to be appropriate, based on the understanding that the magnitudes of applied aero forces for a given condition generally decrease as the order of the characteristic frequency increases.

Turning to the study of transmission of forces into the nonrotating frame, the question became whether or not given (accepted) frequency components of blade forces in the rotating frame were being properly interpreted within the nonrotating frame. This problem is simply one concerning trigonometric conversion of the assumed accepted forces in the rotating frame. The FFT was needed once again, this time to decompose the forces in the nonrotating frame which were calculated by the blade-element rotor model. Once decomposed, the components were checked to see if they properly corresponded to the known characteristics of the forces generated in the rotating frame. Once again, these results were found to be satisfactory.

This frequency content analysis showed that the baseline blade-element rotor model appeared to be:

1. Generating proper frequencies (and probably magnitudes, also) of force components of the blades in the rotating frame.

2. Passing proper frequencies into the nonrotating frame.

This study was important in establishing accuracy and confidence in the baseline blade-element rotor model before proceeding to the task of optimizing of this model for real-time solution applications.

Defining Benchmark Characteristics

In order to determine how the predictive capability of any simplified version of the blade-element rotor model is influenced by candidate model simplifications, some benchmark is needed against which the results of the simplified models can be compared. A relatively unsimplified version of the baseline blade-element rotor model, coupled with the remainder of the helicopter as modeled by UHM, was used to generate "ideal" or "expected" trim and transient aircraft characteristics to serve as the benchmark results (or truth cases) for the sensitivity analysis. The blade-element rotor model representation selected to generate the benchmark results employed the following values of key rotor model parameters:

- Actual number of blades

- 10 segments (elements) per blade

- 2 degree rotor azimuth advance interval

- 1:1 ratio of rotor solutions per fuselage solution

Other blade-element rotor model characteristics selected for the benchmark model representation included yawed (radial) flow at the blade segments during aerodynamic force calculations, and an unfanned segment advance positioning method during rotor model solution corresponding to a particular rotor azimuthal position.

During the course of this study, experiments were conducted to determine if a larger rotor azimuth advance angle between successive discrete blade-element rotor model solutions could be used. This was desired because of the large number of benchmark cases being experimented with, and the relatively long computational time required to complete each benchmark case using a two-degree azimuth increment. It was discovered that an eight-degree azimuth increment provided essentially identical trim and transient

rotor and fuselage characteristics. A number of the benchmark cases were then generated using this larger azimuth increment, being confident that the same results would have been produced with the two-degree increment.

Evaluation Criteria

Under the two-prong approach for analyzing the impact of variation in values of the parameters associated with the blade-element rotor model, first rotor force and blade angular displacements would be scrutinized to identify anomalies which might ultimately create inaccuracies in the helicopter simulation. The second step, and perhaps the more pragmatic test, would survey the impact of the rotor modeling variations on the fuselage trim and response characteristics. Acceptance or evaluation criteria had to be defined according to which conclusions of the sensitivity analysis could be drawn in judging the relative accuracy of the net vehicle simulation, in comparison to the benchmark cases. This approach keeps in mind the goal of fidelity in modeling pilot perceivable vehicle characteristics of the simulator used as a flight training device.

Since the step involving analysis of the impact of rotor model input parameters on net vehicle characteristics would involve the investigation of both trim and transient flight conditions, evaluation criteria for both of these flight conditions seemed necessary. However, during the course of the study it was discovered that the key aircraft trim parameters (pitch, roll, and sideslip angles, and cockpit control positions) exhibited virtually no variation from the benchmark cases as values of blade-element rotor model input parameters were varied during the sensitivity analysis. Therefore acceptable tolerances of the key parameters of the trimmed flight condition tests were never formally defined.

Rotor model variations had a very distinct impact on the transient or dynamic response characteristics of the simulated helicopter. Therefore the parameters which best illustrated such differences in dynamic response of the vehicle, and the amount by which these parameters must differ from the corresponding benchmark solution in order to deem the test case as unacceptable in maintaining the accuracy of the benchmark, were established. Controllability parameters were selected as most practical to meet this application. Five such controllability parameters were selected:

1. Maximum angular acceleration

2. Time to achieve maximum angular acceleration

3. Maximum angular rate

4. Time to achieve 63% of maximum angular acceleration

5. Fuselage attitude at one second

Controllability tests measure these responses as resulting from step control inputs, preferably with a well-defined maximum rate being evident before the aircraft must be recovered from the step disturbance. Although the transient responses of this study were gathered for pulse control inputs, these controllability parameters can be used in the evaluation because over a short period of time (i.e., before the return portion of the pulse input occurs) the aircraft could be considered as responding to a step input.

Table I presents the threshold or tolerance criteria established in this study to define an acceptable level of fidelity between the results obtained from the simulation with rotor model variations, when compared to the benchmark results. The sensitivity analysis comparison task is very similar in approach to the comparison task undertaken when validating the flight characteristics of a simulator with test data of the corresponding aircraft. Therefore, it was decided to draw on the methodology of simulator validation via correlation with flight tests data, to suggest tolerance values which would be applicable to the task of evaluating the impact of model modifications on flight characteristics predicted by the model. Table I also presents examples of the tolerances used by various sources[4,5] for validation of helicopter simulators with test data. In general, the blade-element rotor model sensitivity analysis threshold levels were selected at about one-half the average tolerance values preferred for simulator validation with test data. These tighter requirements were selected because the sensitivity analysis compares math models to math models, and need not account for measurement errors, or for variation in individual aircraft characteristics necessary when comparing model characteristics to actual aircraft characteristics.

Sensitivity Analysis Overview

The Sikorsky UH-60A Black Hawk was selected as the specific helicopter to be simulated during this blade-element rotor model sensitivity analysis (including the introductory rotor power spectrum analysis). The UH-60A was selected because it is a relatively modern helicopter, it employs the familiar articulated rotor hub type, and it is described rather thoroughly in today's literature. Also, Link engineers are intimately familiar

Table 1 Tolerances of Sensitivity Analysis Evaluation Parameters, Compared to Other Sources

HELICOPTER CONTROLLABILITY PARAMETER	CAE-LINK BLADE-ELEMENT SENSITIVITY ANALYSIS	U.S. ARMY UH-60A SIMULATOR SPEC. 223-1152E	FAA HELICOPTER SIMULATOR QUALIFICATION AC 120-XX
Maximum angular rate	±5%	±15%	±10%
Time @ 63% maximum angular rate	±10%	±15%	−
Attitude @ 1 second	±5%	±10%	±10%
Maximum angular acceleration	±5%	±15%	±10%
Time @ maximum acceleration	±10%	±15%	−

with this aircraft, having produced the training simulator for the U.S. Army. Most of the vehicle flight-related characteristics modeled by the UHM version of the simulation were drawn from the flight mathematical model of the Army training simulator (which in turn was based on descriptive data from the airframe manufacturer). The rotor airfoil section aerodynamic coefficient data needed to model the UH-60A rotor system was available.[2]

It was not the objective of this study to validate the absolute fidelity of the helicopter simulation incorporating the blade-element rotor model via a flight test data correlation exercise. However, it was deemed beneficial to first insure that the newly developed engineering model of the rotor realistically predicts rotor characteristics. Trim and transient aircraft flight characteristics as predicted by the UHM with the Link baseline engineering blade-element rotor model were compared with the trim and transient flight characteristics described in Kaplita's report[6], which presents analytic predictions of UH-60A characteristics. The UHM simulation with the blade-element rotor model in its benchmark configuration compared well with Kaplita's data.

For the sensitivity analysis, the UH-60A operation was simulated at nominal rotor speed (258 rpm), aft center-of-gravity location (352 fscg), and at a density altitude of 2000 ft. A high gross-weight value (18,000 lb) was modeled to accentuate rotor trim and transient characteristics. All sensitivity analysis simulation cases were modeled with no stability augmentation in use for the UH-60A. Stabilator incidence was fixed throughout simulated transient maneuvers to better isolate cause-and-effect characteristics to the main

rotor model. The simulation employed a Bailey coefficient representation of the tail rotor.

The sensitivity analysis trim test case flight conditions were done in hover and at level flight at 100 and 130 knots airspeed. The bulk of the transient test cases were lateral or collective stick pulse inputs initiated at 100 knots level flight. Collective and lateral disturbances were expected to create the largest (i.e., most observable) rotor and fuselage responses, respectively. Over 220 cases were run during the course of the sensitivity analysis.

Sensitivity Analysis Results

Number of Blades

A rotor system may be represented by a blade-element rotor model which models fewer (or more) rotor blades than there are in the actual rotor. This can be accomplished by first solving the rotor force generated by the sum of the reduced number of blades, and then scaling this total force by the ratio of actual number of blades to simulated number of blades before proceeding to transfer the effects of the rotor forces to the fuselage. This method works well, even in modeling transient rotor response, for aperiodic rotor characteristics, but introduces inaccuracies regarding periodic characteristics. The sensitivity analysis considered simulation of the rotor system with three, four and seven blades. (The UH-60A has four main rotor blades.)

Fig. 1 illustrates the effect due to number of blades. It shows the aircraft roll response following a lateral control pulse input for the UH-60A helicopter modeled with both the actual number of blades (four)
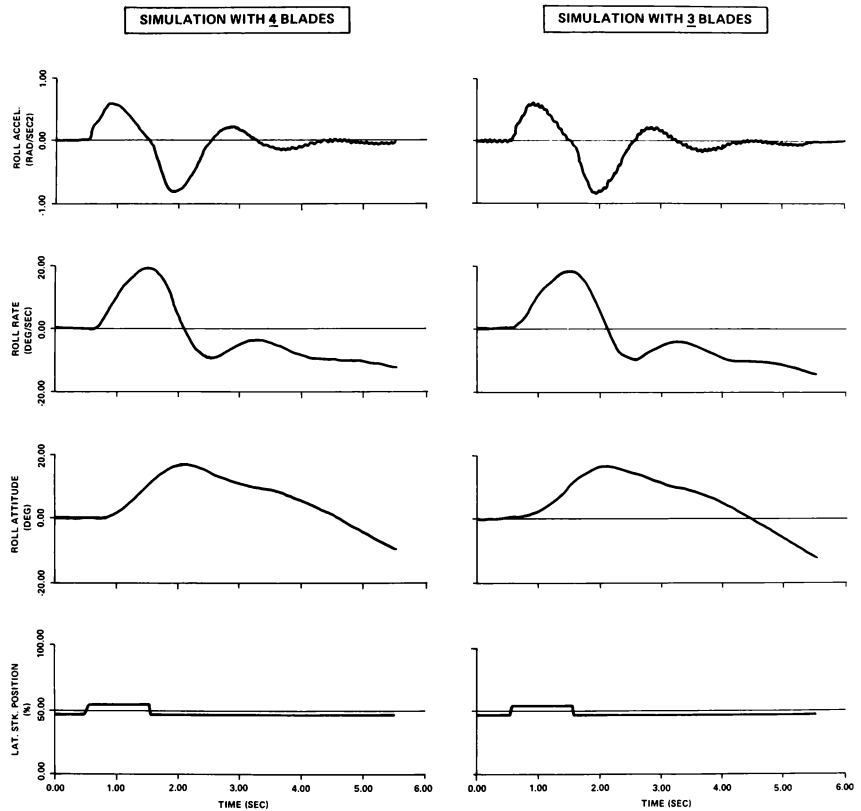
Fig. 1 Effect of Number of Simulated Blades on Helicopter Response

and a reduced number of blades (three). In general, the overall aircraft response is essentially the same for these two models because the aperiodic rotor force is most important in dictating the fuselage response, and this aperiodic rotor characteristic does not suffer significantly when solved by the rotor model represented by a reduced number of blades. Notice in the roll acceleration traces of Fig. 1 that when four blades are modeled, a 17.2 Hz oscillatory characteristic appears in the acceleration time history, whereas when three blades are modeled the oscillation in acceleration is at a frequency of 12.9 Hz. These frequencies reflect the physical filtering characteristic of helicopter rotors: periodic rotor forces generated in the rotating frame which are harmonics of the rotor rotational frequency cancel when passed to and summed in the nonrotating (fuselage) reference frame – with the exception of harmonics of the product of number of blades and rotor speed. (The rotor speed of the simulated UH–60A is 4.3 Hz.) The transfer by the model configuration using three blades of the 12.9 Hz oscillatory component of rotor force from the rotating frame to the nonrotating frame (and the failure to transfer any 17.2 Hz component) is rigorously incorrect. However, since the 12.9 Hz and the 17.2 Hz components have very little power compared to the nonoscillatory component of rotor force, there is very little difference in impact on net fuselage response resulting from the influence of either the 12.9 Hz or 17.2 Hz force component.

The preceding discussion applies to free or natural rotor response, whereby the rotor is excited by some aperiodic disturbance and responds primarily to forces generated at frequencies which are harmonics of rotor speed. Now consider the condition whereby the rotor is excited by some selected sinusoidal control input. Components of rotor forces in the rotating frame which are not harmonics of the rotor speed (or the subset: harmonics of the number of blades times rotor speed) are passed to the nonrotating frame, but can be substantially attenuated depending on the proximity of the nonharmonic frequencies to harmonic frequencies. The attenuating effect of the modeled natural rotor filter on nonharmonic frequencies will differ depending on the number of rotor blades represented within the blade–element rotor model. Frequency domain analysis of test cases with different sinusoidal control excitations showed this to be true. Only when the model recognizes the same number of blades as exist in the actual rotor can we expect the model to produce the same filtering characteristics as the actual rotor.

It is currently recommended that the optimized blade–element rotor model designed for real–time simulator application employ the actual number of rotor blades. This will insure minimal distortion of the magnitudes of rotor forces transferred from the rotor to the fuselage under all circumstances. This recommendation is made even though analysis of controllability–type flight conditions indicates that the use of three (or seven) blades does not significantly modify the aircraft response in comparison to the response produced for the same step or pulse control inputs when four blades are modeled. (Trim characteristics were not significantly impacted by variation in blade number during the sensitivity analysis.) It is the low-frequency, periodic pilot control inputs which are probably most susceptible to attenuation by a rotor model employing a number of blades different from the number on the actual helicopter. Rotor force components of low frequency most likely contain more power than higher–frequency components and therefore would have greater influence on net aircraft response.
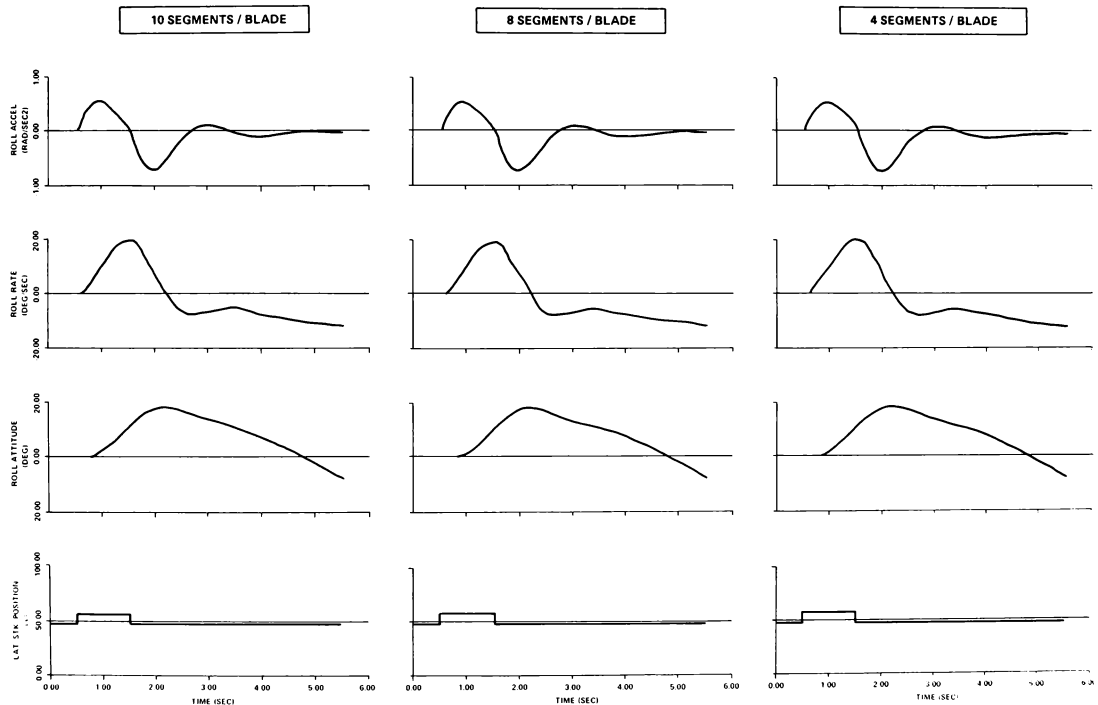
Number of Segments

Model conditions with eight segments and four segments per blade were studied in comparison with the benchmark condition of ten segments per blade. The segment size and position were selected according to the traditional method of representing equal elemental disk areas or annuli. This technique weights finer resolution of detail of the segments outward from the center of rotation where higher dynamic pressures are encountered.

Fig. 2 summarizes the findings of this study. The number of segments per blade in the blade–element rotor model can be reduced to (at least) as little as four without noticeably impacting the trim and transient flight characteristics of the helicopter when compared to the benchmark model configuration.

Azimuth Advance Interval

The rotor azimuth increment modeled between successive discrete rotor solutions of the blade–element model was tested at values of 8, 12, 20, and 30 degrees. Compared to the modeled flight characteristics obtained using a benchmark configuration of a 2–degree azimuth increment within the baseline blade–element rotor model, negligible differences result up to an azimuth increment of 20 degrees. When an increment of 30 degrees is used, differences in modeled flight characteristics occur which cannot be judged as negligible. This is somewhat apparent when studying trim characteristics, but is more obvious

Fig. 2 Effect of Number of Segments Per Rotor Blade on Helicopter Response

when examining transient response cases. Somewhere between 20 and 30 degrees, discernible differences in modeled characteristics begin to occur. The conservative approach would then define 20 degrees as this limit. Table 2 provides examples of the cases studied which defend this conclusion. The evaluation criterion used in this analysis is that discussed earlier regarding controllability-type parameters.

When the number of segments per blade was considered in the sensitivity analysis simultaneously with the azimuth increment, there were no changes in the conclusions. That is, even as the number of segments was reduced from ten to eight, and to four, substantial variation in flight characteristics was not observed as azimuth increment was also increased, until the 30 degree cases were studied. This indicates that segment number can be reduced to four, with a 20-degree azimuth increment, without degrading the accuracy provided by a 10-segment/20-degree-azimuth configuration, or even compared to that provided by a 10-segment/2-degree-azimuth model configuration.

Rotor Filter

Some users of blade-element rotor models within helicopter simulators prescribe the application of a first-order, low-pass filter between the rotor model and the fuselage (net vehicle) model. In particular, this filter is applied to the total rotor forces and moments expressed in a nonrotating rotor reference frame before being transformed into a fuselage or body reference frame. The intent of this filter is to remove high-frequency content from the rotor forces and moments which might introduce instabilities within the discrete model solution, or to eliminate unwanted higher-frequency characteristics from being introduced into system components of the simulator (e.g., into the motion system). It may even be argued that such a filter has a physical equivalence in the actual helicopter, representing (for example) vibration absorption characteristics of the mechanical components of the rotor head.

The impact of such a rotor filter in the helicopter simulation was investigated within the sensitivity analysis. A first-order, low-pass filter with a time constant 0.3 seconds was modeled, acting where rotor forces are transferred to the fuselage. Experiments running identical transient response simulations with and without the inclusion of such a rotor filter showed that the filter introduced false low-frequency content in the net vehicle response characteristics. For example, for lateral cyclic stick pulse input cases, the roll response of the UH-60A exhibited a characteristic response frequency of about 0.7 Hz in roll acceleration and roll rate. The magnitude of this response was significant. No 0.7 Hz roll characteristic was demonstrated within the benchmark cases for which no rotor filter was included.

Table 2 Examples of Sensitivity Analysis Results for Segment Number and Azimuth Angle Increment Test Cases

EVALUATION PARAMETERS	BENCHMARK (10 seg, 2 deg) RESULTS	ACCEPTABLE RANGE	TEST CASES					
			10 seg, 12 deg	10 seg, 20 deg	10 seg, 30 deg	8 seg, 12 deg	4 seg, 8 deg	4 seg, 30 deg
Time at maximum roll acceleration (seconds)	0.35	0.31 to 0.39	0.36	0.35	0.36	0.35	0.34	0.36
Maximum roll acceleration (rad/s^2)	0.55	0.52 to 0.58	0.56	0.56	0.51	0.56	0.55	0.60
Maximum roll rate (deg/s)	20.0	19 to 21	20.0	19.7	19.2	20.0	20.2	19.4
Time at 63% maximum roll rate (seconds)	0.57	0.51 to 0.63	0.56	0.55	0.52	0.55	0.56	0.53
Change in roll attitude at 1 second (degrees)	10.0	9.5 to 10.5	10.2	10.4	10.9	10.4	10.1	10.9

When the high-bandwidth blade-element rotor model method was employed in the helicopter simulation, the benchmark cases demonstrated no computational instability problems in vehicle accelerations that are introduced by the vibratory content of the rotor. The sensitivity analysis results clearly indicate a preference to refrain from providing any filtering between the interface of the blade-element rotor model and the dynamic fuselage model. Should it be desirable to repress any N/rev harmonic frequencies of fuselage acceleration introduced from the rotor from entering any real-time simulator systems (e.g., motion, visual, hardware components), these frequency components should be filtered prior to entry into the affected systems, but should not be filtered immediately at their calculation within the fuselage kinematical solution.

Rotor:Fuselage Solution Ratio

Circumstances may sometimes dictate that a helicopter simulation solve the blade-element model more frequently than the fuselage or general vehicle dynamic model is solved. This may arise if a new rotor model is being retrofitted into an existing older helicopter simulator. Or perhaps computer resources or computational requirements may demand a maximum of time-savings be achieved, and this might be accomplished by minimizing the number of solutions of the net vehicle equations of motion.

Part of the sensitivity analysis was devoted to studying the impact of changing the ratio of rotor-to-fuselage solutions within the simulation. Preliminary analysis tested rotor:fuselage solution ratios of 193.5:193.5, 193.5:19.35, and 193.5:9.675. (193.5 rotor solutions per second corresponds to a rotor azimuth advance of 8 degrees per discrete solution for the UH-60A.) The 193.5:193.5 (i.e., 1:1) ratio served as the benchmark. Results indicated the 193.5:19.35 ratio (i.e., "10:1") produced acceptable results (according to the established training simulator evaluation criteria), while the simulation with a rotor:fuselage solution ratio of 193.5:9.675 (i.e., "20:1") was unacceptable. But such generalization may not be useful. Solution ratios of 90:30 versus 180:60 might produce different results (due to the different frequencies which could be accurately passed through the fuselage model computed at 30 Hz versus 60 Hz), yet these two examples are both 3:1 ratios between the rotor and net vehicle solutions within the simulation. Also, this particular sensitivity analysis study is further complicated by the use of filters within the simulation, whether nested within the rotor model to isolate and remove high-frequency rotor content to prevent it from being aliased, or placed between the rotor and the fuselage to attenuate rotor forces and moments more generally as they are transferred to the fuselage.

For these reasons this study did not attempt to draw any generalized conclusions regarding the impact of rotor:fuselage solution ratio at this time. Perhaps the best approach to addressing this question is to specifically test such solution ratios of interest for a particular application, as applied to the specific form of the simulation math model which has been derived for the task at hand.

Fanned Segments

Fig. 3a illustrates the normal location of blade segments during solution of the blade-element rotor model, for two consecutive calculations of the entire rotor state. If a staggered or fanned location of segments between net azimuth angle change from one rotor solution to the next were used, as illustrated in Fig. 3b, could accuracy which may have been lost due to some modeling simplications be restored to the rotor solution? For example, if azimuth increment update between consecutive rotor calculations were preferred to be large, could this fanned method of solution of individual segment aerodynamics result in a better average of net rotor characteristics, which more closely matched benchmark characteristics (established at a small azimuth advance increment) than would a solution at this large increment without fanning the azimuth positioning of segments? To test this concept and answer such questions, the baseline blade-element rotor model program was modified to provide the option of calculating such a fanned solution.

Following this program modification, sensitivity analysis cases were run to study the impact of "fanning" on the accuracy of the model. For these tests, ten segments per blade were modeled, and azimuth advance increment was varied, up to 50 degrees, using the fanned approach. Such model configurations were tested for various flight conditions, studying both trim and transient response in the time domain. For a particular azimuth advance increment greater than the value selected for the benchmark configuration (i.e., two degrees), the solution using the fanned method and the solution using the standard method were both compared to the corresponding "truth" solution from the benchmark configuration. These comparisons were studied to determine if the solutions obtained using the fanned method more closely matched the "truth" solutions than did the solutions using the standard method.
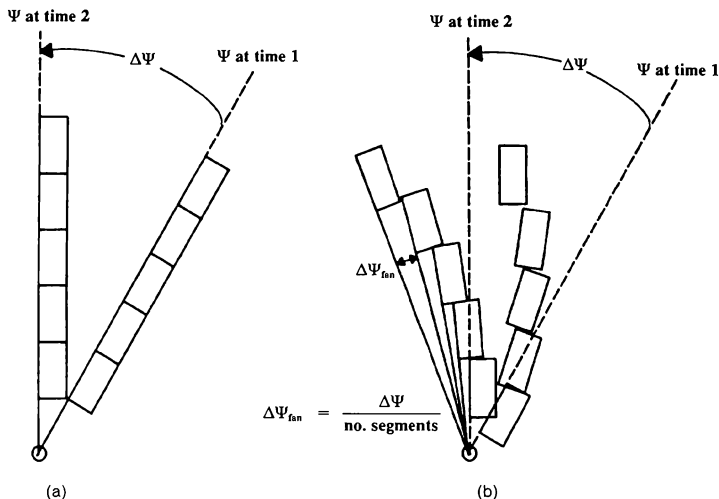
Fig. 3 Traditional vs. Fanned Distribution of Blade Segments at Two Consecutive Rotor Solutions

Both trim and transient response characteristics clearly indicated that the fanned method did not improve accuracy of solution. The added complication to the model required to model this fanning effect will not restore accuracy lost when making some modeling simplification (such as use of a comparatively large azimuth advance interval).

Yawed Flow Models

This sensitivity analysis task was concerned not with variation of an input parameter value within a given model, but instead (like the fanned segment task) was concerned with the impact of variation of the model itself on the solutions obtained from the blade-element rotor model. Howlett[2] describes a methodology for approximating effects due to yawed (i.e., blade spanwise) flow components of velocity at the rotor. (This methodology is designed to work in conjunction with basic two-dimensional airfoil data which does not rigorously recognize true spanwise flow effects.) This yawed flow consideration was included in the Link baseline blade-element rotor model. A portion of the sensitivity analysis addressed the question of how the flight characteristics of the helicopter would

be affected if the yawed flow were omitted. Two different techniques for excluding yawed flow and resorting to more simple two-dimensional characteristics were tested. While transient response characteristics were essentially identical between the baseline model and the two "no yawed flow" versions, there were perceptible differences in trim characteristics between the three model configurations. (Pitch attitude differed by about 0.5 degrees, and longitudinal cyclic stick varied by as much as two or three percent.) Actually, this variation was seen between the two "no yawed flow" configurations. The differences in these two versions of the math model were arbitrary and small; they reflected two equally "sophisticated" means of removing the third dimension from the flow field about the rotor blade segments. One approximation was not more exact than the other.

The results of this portion of the sensitivity analysis show that the rotor model (and the resulting impact on fuselage trim and response characteristics) can be noticeably affected by minor changes in the details of the math model. Sometimes such choices of detail are arbitrary. Under such circumstances this provides a perplexing choice to the engineer, since otherwise

quivalent models can produce non-negligible differences in the resulting prediction of aircraft flight characteristics. This observation is quite important, not so much in the context of the blade-element method sensitivity analysis, but more so in the understanding of modeling limitations and the degree of fidelity ultimately achievable when comparing model predictions with test data of the actual aircraft.

Conclusions/Recommendations

1. Table 1 presents recommended controllability-type test criteria for evaluating the impact of blade-element rotor model input parameter values and modeling considerations on solution accuracy within the context of helicopter training simulator. Transient response time history test cases demonstrate the impact of changes in the blade-element rotor model within a sensitivity analysis; trim cases demonstrate little or no information regarding the tested model simplifications.

2. Even though the adequacy of a blade-element rotor model within the context of a helicopter training simulator is best judged by studying net vehicle response, it pays to first explicitly study modeled rotor characteristics. This proves to be particularly important to avoid or prevent aliasing problems and the severe repercussions that they can create in net vehicle response.

3. Although aliasing problems should be addressed during initial blade-element rotor model development stages, this problem must be continuously observed and studied during all stages of the development of the helicopter simulator math model. Solutions to the ubiquitous and potentially serious aliasing problems do not lie solely in simple modifications to the original isolated rotor model, but in the basic structure and characteristics of the complete helicopter math model.

4. Houck's findings[1] are still basically pertinent today. Stronger emphasis on modeling within the simulator the same number of rotor blades as the actual aircraft is recommended. Greater emphasis on considering the frequency content of rotor characteristics within the sensitivity analysis methods is also recommended, to better understand the potential aliasing problem.

5. A rotor azimuth advance angle of approximately 20 degrees was determined to produce no discernible loss in accuracy of the net vehicle response characteristics for a helicopter simulator employing the blade-element rotor model tech-nique. This azimuth advance increment (in conjunction with the helicopter rotor speed) dictates the required frequency of solution of the blade-element rotor model within the simulation.

6. In general, gross filters acting between the rotor model solution of forces and moments produced at the rotor head, and the transfer of these forces and moments to the airframe, for the purpose of eliminating high frequency content, are not needed. Use of such filters may actually introduce false frequency content into the net vehicle kinematical responses.

7. Generalization of sensitivity analysis results from studies conducted modeling a particular helicopter may not be recommended for application to some different helicopter simulation. Instead, a standing design tool is recommended through which a sensitivity analysis could be quickly and easily conducted early in the design stage for each particular helicopter simulation application for which model simplifications are sought. Such a design tool should follow a two-pronged analysis approach, studying rotor characteristics first, followed by net vehicle dynamic and kinematic characteristics. The rotor analysis of such a study should employ some frequency domain analysis methods to complement studies done in the time domain.

References

1. Houck, J.A., "Computational Aspects of Real-Time Simulation of Rotary-Wing Aircraft," NASA Contract Report 147932, May 1976.

2. Howlett, J. J., "UH-60A Black Hawk Engineering Simulation Program: Volume I – Mathematical Model," NASA Contract Report 166309.

3. McFarland, R. E., "Spectral Decontamination of Real Time Helicopter Simulation," presented at the AIAA Flight Simulation Technologies Conference, Niagara Falls, NY, June 1983.

4. Anon., "Production Specification for UH-60 Black Hawk Helicopter Synthetic Flight Training System", 223-1152E, 8 January 1983, Naval Training Equipment Center, Orlando, Florida.

5. Anon., "Rotorcraft Simulator, Visual and Motion System Evaluation", FAA Draft Advisory Circular 120-XX, June 6, 1985, Federal Aviation Administration, Atlanta, Georgia.

6. Kaplita, T. T., "UH-60A Black Hawk Engineering Simulation Model Validation and Proposed Modifications," NASA Contract Report 177360, July 1985.

REAL-TIME NETWORKING FOR A DISTRIBUTED
AIR COMBAT SIMULATION

Matthew R. Lobeck
Senior Hardware Design Engineer,
Project Lead Engineer

Michael T. McAuliffe
Specialist Software Design Engineer

The Integrated Technology Development Laboratories
Boeing Defense & Space Group
Military Airplanes Division
Seattle, Washington

Abstract

Boeing Defense & Space Group Military Airplanes Division Integrated Technology Development Laboratories (ITDL) has developed a real-time network for its HIFEN2 distributed air combat simulation. The backbone of the HIFEN2 network is Proteon's ProNET-80 local area network. This paper will include an overview of the HIFEN2 distributed air combat simulation, and will examine the stringent requirements that HIFEN2 imposes on network interfaces. The paper will also show that commercially available ProNET-80 interfaces were unable to meet HIFEN2 requirements, and will show that the ITDL was able to adapt ProNET-80 for HIFEN2 by developing custom host interfaces to replace Proteon's host interfaces. A custom host interface has already been developed for Gould computers, and a custom host interface is currently being developed for Silicon Graphics and other 9u VMEbus host computers. The development, functionality, and application of these custom interfaces will be examined.

Introduction

The Integrated Technology Development Laboratories (ITDL) is a Boeing Defense & Space Group Military Airplanes Division facility dedicated to the research, development, and evaluation of advanced aircraft technologies and systems. Individual laboratories within the facility are dedicated to the research and development of diverse aircraft technologies, including advanced pilot-vehicle interfaces, avionics, and flight controls. Additionally, the facility houses three high-fidelity flight simulator domes which are used to integrate and demonstrate the advanced technologies and

systems developed in the ITDL and other areas of Boeing.

Adequate test and evaluation of a new aircraft system cannot always be accomplished with a single vehicle simulation. In such cases, a multiple engagement air combat simulation that accommodates interactive combat between multiple manned and unmanned simulators is required. To provide such a capability in the ITDL, a distributed, real-time, multiple participant, air combat environment simulation called High Fidelity Environment Two (HIFEN2) has been developed.[1] HIFEN2 provides aircraft systems designers with the ability to test and evaluate their advanced aircraft technologies and systems in realistic simulated air combat.

HIFEN2, like any multiple engagement simulation, must provide a way for multiple single vehicle simulations, in this case executing on multiple host computers, to communicate data about their common environment to one another. During a HIFEN2 simulation, each participant simulations must pass data to all other participants identifying its current location and sensor emissions. Additional inter-participant communications is required for simulation synchronization. For HIFEN2, this real-time inter-participant communication is accomplished via a high-speed data network that is called the HIFEN2 Environment Network. The HIFEN2 Environment Network has been implemented with Proteon's ProNET-80 local area network.

This paper will show how the ITDL is using ProNET-80 for HIFEN2 networks. An overview of HIFEN2 will be presented, with specific emphasis on inter-participant communications over the HIFEN2 Environment Network. Network loading will be

examined, along with the stringent requirements that HIFEN2 places on host computer network interfaces.

It will be shown that Proteon's off-the-shelf ProNET-80 network interfaces were unable to meet HIFEN2 requirements. Although the network control logic (CTL) portion of Proteon's interfaces are capable of the ProNET-80 network's full 80 Mbps throughput, the host specific board (HSB) portion of the Proteon interfaces are not optimized to support this bandwidth into a host computer. To adapt ProNET-80 for HIFEN2, the ITDL has developed custom ProNET-80 interfaces for HIFEN2 host computers. These custom interfaces utilize Proteon's ProNET-80 CTL board and ITDL designed host specific logic. A custom ProNET-80 interface called the P80GSI has been developed for Gould host computers, and is currently supporting HIFEN2 simulations. A custom ProNET-80 interface called the P80VME is now being developed for Silicon Graphics and other 9U VMEbus computers.

HIFEN2 Overview

Figure 1. shows the configuration of a typical HIFEN2 simulation. In this example a total of fifteen simulated participants are hosted on five computers that are connected to a common Environment Network. The simulated participants in this example include two high-fidelity manned dome participants hosted on Gould 9780 computers, three manned medium-fidelity Non Visual Piloted Crewstations (NVPC) hosted on Silicon Graphics IRIS 4D/120 workstations, and ten Computer Controlled Aircraft (CCA) hosted on a single Gould 9780.

The architecture of HIFEN2 is synchronous. All participant simulations synchronize their execution to a synchronization signal from a HIFEN2 master participant. The master participant is the fastest (shortest frame time) participant in a HIFEN2 simulation, and all other participants typically execute at the same frame rate or at a frame rate that is a multiple of the master participant's frame rate. Each time the master participant begins a new processing frame it transmits a synchronization signal to all other HIFEN2 participants over the Environment Network. This synchronization signal is used by the other participants to determine when to start their own processing frames.

The architecture of HIFEN2 is also distributed. A dedicated HIFEN2 computer is not required for a simulation, as each host computer performs the HIFEN2 calculations for its hosted participants. The HIFEN2 calculations that each computer must perform

include the calculation of relative geometry and vehicle signatures for each hosted participant. Once per frame, via the Environment Network, each participant sends a vehicle state data packet to all other participant
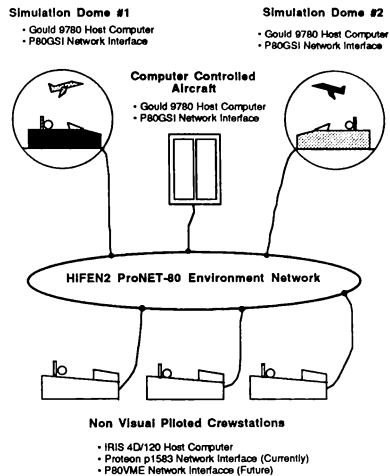

Figure 1. HIFEN2 Multiple Participant Simulation

simulations. This vehicle state packet contains data that describes the current state of the participant, including current location and orientation, and data that identifies the current sensor emissions of the participant and the vehicle signatures (radar cross section and infrared) with respect to all other participants.

HIFEN2 Participant Processing

Figure 2. shows a typical processing frame for a HIFEN2 participant simulation, and highlights the HIFEN2 communications that occurs during a frame.

A participant's processing frame is initiated by the reception of a HIFEN2 synchronization packet. The synchronization packet is a broadcast packet that is transmitted onto the Environment Network by the HIFEN2 master participant each time it begins a new frame. Since the synchronization packet is a broadcast

packet it is received by all computer nodes on the Environment Network. All participants except the master use the reception of the synchronization packet or multiple synchronization packets as a signal to begin a new processing frame.
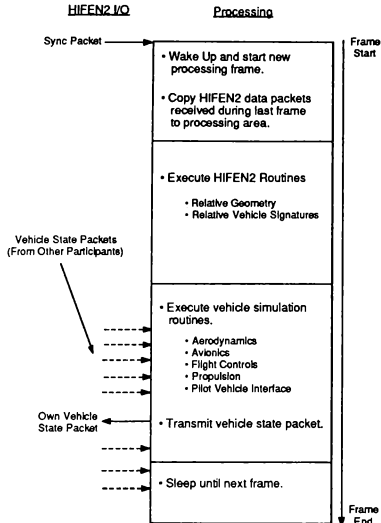
HIFEN2 I/O   Processing

Sync Packet ────────────────────────────── Frame
                  • Wake Up and start new   Start
                    processing frame.

                  • Copy HIFEN2 data packets
                    received during last frame
                    to processing area.

                  • Execute HIFEN2 Routines

                        • Relative Geometry
                        • Relative Vehicle Signatures

Vehicle State Packets
(From Other Participants)

                  • Execute vehicle simulation
                    routines.

            ─────┤   • Aerodynamics
            ─────┤   • Avionics
            ─────┤   • Flight Controls
            ─────┤   • Propulsion
            ─────┤   • Pilot Vehicle Interface

Own Vehicle ◄─────┤
State Packet   • Transmit vehicle state packet.

            ─────┤

            ─────┤ • Sleep until next frame.
                                              Frame
                                              End

Figure 2. HIFEN2 Participant Frame

At the beginning of a frame, after a synchronization packet has been received, all vehicle state packets that were received from other participants during the previous frame are copied to a predefined memory area for processing later in the frame. Next, HIFEN2 routines are executed to compute relative geometry and relative vehicle signatures. Finally, the vehicle simulation routines are executed. This processing includes vehicle aerodynamics, avionics, flight controls, propulsion and pilot vehicle interface processing. A vehicle state packet describing the current state of the participant is then transmitted onto the Environment Network, and the participant then sleeps until the start of its next frame.

HIFEN2 Communications

All HIFEN2 inter-participant communications is accomplished over the HIFEN2 Environment Network. The Environment Network has been implemented with Proteon's ProNET-80 local area network. Each computer node that hosts HIFEN2 participants is configured with a ProNET-80 hardware interface board that enables its hosted participants to send and receive ProNET-80 packets containing HIFEN2 data via the Environment Network. Currently, both Gould and Silicon Graphics systems host HIFEN2 participants. For ProNET-80 communications, Gould host computers utilize the P80GSI ProNET-80/Gould SelBUS hardware interface that was developed in the ITDL. At this time, Silicon Graphics host computers are equipped with Proteon's p1583 ProNET-80 hardware interface.[2] In the future, Silicon Graphics hosts will use the P80VME ProNET-80 VMEbus hardware interface now being developed in the ITDL.

Why ProNET-80 ?

There are several reasons why ProNET-80 was chosen for HIFEN2 communications. First, the ITDL had previous experience with Proteon's ProNET-10 network. ProNET-10 is functionally the same as ProNET-80, and upgrading to ProNET-80 enabled the ITDL to obtain a factor of eight improvement in network bandwidth without switching to a completely new network architecture. Additionally, since ProNET-80 is a token passing network, its performance is more deterministic than that of a network that allows collisions, such a Ethernet. This determinism is important for flight simulation applications where all network communications for a given simulation frame must complete before the end of that frame. If a network does not allow collisions, and if the delay through each network node can be calculated, the total time for a frame's inter-participant communications can be calculated. It can then be verified that all communications will indeed complete within the allocated frame time. A third reason for the selection of ProNET-80 for the HIFEN2 communications network is that it provides a fiber-optic link between each node. This supports the security requirements of the ITDL.[3]

ProNET-80 Network

Proteon's ProNET-80 local area network supports inter-computer communications at 80 Mbps. Systems which utilize ProNET-80 require a hardware link between the ProNET-80 network and the host system. This hardware link, called a network interface, performs the

actual reception and transmission of ProNET-80 data packets on behalf of the host system. Some of the specifics of ProNET-80 are detailed in the following paragraphs:

ProNET-80 is a single token passing ring. Any interface which is connected to the ProNET-80 network and wishes to transmit a data packet onto the network must wait for the token. Transmission involves detection of the token, insertion of the new data packet, creation of a new token, and removal of the packet after it has circulated around the ring.

A message on the ProNET-80 network is a series of modulated bits with 4-into-6 4b/6b balance encoding. At the beginning of a packet there is a Beginning Of Message (BOM) flag, and there is also another flag at the of the packet called the End Of a Message (EOM). A message is normally followed by another message unless its is the last one put onto the network, in which case it is followed by the token.

When a ProNET-80 interface is ready to receive packets, it looks for a BOM flag, and when it sees a BOM it compares the destination node address of the packet with its own node address. If the addresses are the same, or if the destination node address is the broadcast address, the entire packet (up to the EOM flag) is accepted by the interface.

Environment Network Loading

During a HIFEN2 simulation, the network loading of the HIFEN2 Environment Network is influenced heavily by the frame-based synchronous nature of HIFEN2. This loading is characterized by alternating periods of intense network activity and periods of minimal or no network activity. Figure 3. shows the loading of the Environment Network during a typical HIFEN2 simulation. The network loading that is demonstrated here is the loading for the fifteen participant simulation shown in Figure 1.

In this example, the two high-fidelity dome participants execute 25 msec frames, the medium fidelity NVPCs execute 50 msec frames, and the low fidelity CCAs execute 100 msec frames. One of the high fidelity dome participants is the master participant, and it issues the HIFEN2 synchronization packet every 25 msec as it begins each new frame. All other participants begin a new frame when they receive the synchronization packet, or after multiple synchronization packets are received if they have a frame time that is longer than that of the master participant.

Each participant transmits a vehicle state packet onto the network near the end of its processing frame. Each vehicle state packet is a maximum size ProNET-80 broadcast packet (1024 words) and requires approximately 410 usec to travel around the ProNET-80 network. These broadcast packets are received by all nodes on the network, and they constitute the majority of the network traffic on the Environment Network.

As Figure 3. shows, the combined loading of the HIFEN2 Environment Network is very uneven because of the frame-based synchronous nature of HIFEN2.
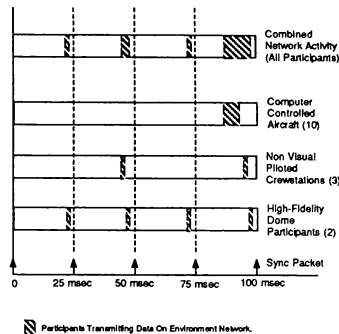

Figure 3. Environment Network Loading

The loading of local area networks in a campus or office file server environment tends to be somewhat distributed over time. HIFEN2 network loading. on the other hand, features alternating periods of peak activity when many back-to-back packets are on the network, and periods when network activity is extremely light. Network activity is very light at the beginning of a frame and extremely heavy near the end of a frame when participants are transmitting their vehicle state packets. Loading also varies from frame to frame because there are some frames when only the highest-rate participants transmit, and other frames when all participants transmit. It has proven difficult or impossible for some HIFEN2 host computers that utilize off-the-shelf ProNET-80 hardware interfaces to receive from the network and copy to host memory all of the back-to-back packets that traverse the network during the I/O peaks that occur at the end of a frame.

Interface Requirements

The ITDL has imposed two stringent requirements on the ProNET-80 network interfaces that host computers use for communications on the HIFEN2 Environment Network. The first requirement is that a network interface must be able to capture from the network every ProNET-80 packet that contains its node address or the broadcast node address in the destination field of the packet header. This requirement that an interface not refuse any data directed to it is imposed for two reasons. First, since the ITDL is a research facility, it is necessary to accurately and thoroughly capture for later analysis all inter-participant communications that occur on the Environment Network during a simulation. In order to satisfy this requirement a HIFEN2 host computer's ProNET-80 interface cannot miss any packets that are transmitted to it. Additionally, the frame times of HIFEN2 participant simulations are very short, and a participant simulation does not have time to retransmit a packet that is missed by another participant, without exceeding its own frame time. Every packet must be received when first sent as the packet will not be sent a second time.

The second requirement imposed on the ProNET-80 network interfaces for HIFEN2 host computers is that they must support a high data transfer rate from the network to host memory. It is not enough for a network interface to be able to copy data from the network to its own internal memory buffers at the full network bandwidth. The interface must also be able to provide that data quickly to the host computer. Large-scale HIFEN2 simulations could utilize much of the bandwidth of the ProNET-80 network, and a host computer's ProNET-80 interface should provide the host simulation with received data at a rate equal to or exceeding the bandwidth of the network. This minimizes the loss of processing time due to data transfer overhead.

ProNET-80/Gould SelBUS Interface

The first multiple engagement air combat simulation, developed in the ITDL was called HIFEN1, and it used Proteon's ProNET-10 local area network for its inter-participant communications network. As it became necessary to support more simulated participants than HIFEN1 could support, because of the relatively low 10 Mbps bandwidth of ProNET-10, it became apparent that a network with a higher bandwidth was needed. The decision was made to migrate to ProNET-80 for the ITDL's second generation multiple engagement air combat simulation, HIFEN2.

To migrate to ProNET-80, a ProNET-80 network interface was needed for Gould simulation host computers. The ProNET-80 p1680 Gould SelBUS Interface is available from Proteon, but did not meet ITDL requirements for a host computer network interface.[4] The p1680 connects to the SelBUS of a Gould computer via the High Speed Data Interface (HSD II). As the bandwidth of the HSD is only 24 Mbps, the Receive Packet FIFO of the p1680 cannot be emptied by the HSD as quickly as it can be filled from the network which has an 80 Mbps bandwidth. During periods of peak loading on the HIFEN2 Environment Network, the Receive Packet FIFO would fill up, causing the p1680 to reject some incoming ProNET-80 packets. This violates the ITDL requirement that a host computer interface must be able to capture all packets transmitted to it with no loss of data.

Since an off-the-shelf ProNET-80 network interface suitable for HIFEN2 was not available, the ITDL engineering staff designed and built the ProNET-80 Gould/SelBUS Interface (P80GSI). The P80GSI meets ITDL requirements for host computer network interfaces. It supports the full 80 Mbps bandwidth of ProNET-80 from the network to its on-board memory, and receives all packets on the network with no loss of data. The P80GSI is accessed as shared memory via the Gould SelBUS, and a Gould host can copy data from the interface at 13 Mbytes/sec.

Architecture

The P80GSI consists of two printed circuit boards. The first printed circuit board, purchased from Proteon, contains the ProNET-80 Control Logic (CTL) that interfaces the P80GSI to a ProNET-80 network. The CTL performs bit-level operations and signal modulation and demodulation for sending and receiving data to and from the ProNET-80 network. The P80GSI utilizes a custom version of Proteon's p1582 CTL80V. At the request of the ITDL, Proteon laid out and manufactured a special version of the p1582 CTL that conforms to the Gould Device Interface (DI) form factor. This custom CTL is called the CTL80B.

The second printed circuit board is the bridge between the CTL logic and the Gould SelBUS. This board was developed in the ITDL, and contains three logic sections: the receive logic and memory section, the transmit logic and memory section, and the SelBUS interface logic. The P80GSI also features a real-time clock with 1 usec resolution that can be read by a host, as well as logic for processing host commands and for providing interface status to a host.

The receive section of the P80GSI contains logic that processes ProNET-80 packets copied from the network by the CTL and memory buffers into which packets are copied. The receive logic examines the header of each received packet, and based upon an identifier in the packet header determines if the packet is to be saved, and if so where in on-board memory the packet is to be stored. On-board memory consists of sixty-four 1024 word receive buffers which are double-buffered. These receive buffers are dual-ported, and are accessed directly by the host system via the SelBUS. At startup a host application supplies the P80GSI with a Receive ID Table that defines the buffer into which each received packet is to be copied. This table also specifies certain packets that when received will result in the P80GSI generating an interrupt to the host system. This feature is used for notifying the host system when a HIFEN2 synchronization packet has been received.

The transmit section of the P80GSI includes the logic that coordinates the transmission of host supplied ProNET-80 packets onto the network. It also includes sixty-four 1024-word transmit buffers that can be accessed by a host via the SelBUS. To transmit a packet, a host writes a transmit packet into one of these buffers. The host then writes a transmit command to the P80GSI that specifies which buffer to transmit. When a transmit command is received from the host, the transmit logic coordinates the transmission of the packet by supplying the CTL with packet data words to send onto the network, and by operating the handshake lines between the CTL and the transmit logic.

The SelBUS interface section of the P80GSI contains the logic that controls host access to the on-board transmit and receive memory buffers.

Using The P80GSI

At the beginning of a simulation run the P80GSI is initialized by the host system. This involves performing a reset of the interface and loading a Receive ID Table that defines the specific ProNET-80 packets that the interface is to receive. All HIFEN2 ProNET-80 packets contain an identifier in the packet header called a content label. When a ProNET-80 packet is received, the P80GSI looks up the packet's content label in the Receive ID table to determine if the packet is one that is to be saved, and if so, which receive memory buffer the packet is to be stored in.

After reset and initialization, the P80GSI begins to copy packets from the network and is ready to respond to any host commands. When a HIFEN2 synchronization packet is received, the P80GSI generates an interrupt that the Gould host uses as a

signal to start a new processing frame. Upon receiving this interrupt, the host operating system wakes up the sleeping simulation task to begin a new frame.

Upon being awakened by the operating system, the first activity that the simulation application performs is to copy from the P80GSI receive buffers the HIFEN2 data packets received during the previous frame. Because the receive buffers on the P80GSI are double buffered, packet reception from the network can continue while the host copies packets from the P80GSI. The P80GSI memory buffers and control registers are memory mapped directly into the address space of the simulation application, so a device driver is not required for board access or control. Rather the simulation application uses its own software routines to write commands and transmit packets to the interface, and to copy received packets from the interface to host memory.

At the end of its frame after its HIFEN2 processing has been performed, the simulation application builds a vehicle state packet. It then copies that packet, and any other packets to be transmitted to other participants, to transmit buffers on the P80GSI. The application then writes transmit commands to the Transmit Command FIFO on the P80GSI to initiate the transmission of the packets.

ProNET-80 VMEbus Interface

The ProNET-80 to VMEbus interface (P80VME) project was started as a result of a study performed in the ITDL on ProNET-80 throughput into a Silicon Graphics (SGI) workstation. The study utilized an SGI 4D/120 workstation, the Proteon p1583 ProNET-80 interface, and a software device driver developed by ALTUM Incorporated in conjunction with the ITDL engineering staff. The study showed that the p1583 was not capable of meeting the throughput requirements that would be placed on it during a real-time distributed air combat simulation. One of the main problems discovered is the high rate of interrupts generated by the p1583 during I/O peaks when multiple back-to-back packets are received. The interface features a two deep packet-status buffer that, when full, rejects all received packets until the buffer is serviced. Each time a new packet is received, the p1583 generates a VME interrupt to the host system. Because of its variable interrupt latency, the IRIX operating system, did not always respond to interrupts quickly enough. In some cases the packet-status buffer would fill up, resulting in the loss of data. This violates a requirement placed on a HIFEN2 host computer interface that all data from every packet must be captured with no loss of data. Another problem found on the p1583 is that the DMA controller

was found to be unreliable. In order to obtain the maximum throughput from the VMEbus into the workstation memory the interface board must be capable of performing VMEbus Block Level Transfers (BLT). Since the p1583 did not meet the requirements of a HIFEN2 host computer interface it was necessary to develop the P80VME.

P80VME Requirements

The requirements for the P80VME conform to the HIFEN2 network interface requirement that an interface must capture all data from the network with no data loss, and to the requirement that an interface must support a high data transfer rate between the interface and its host. The P80VME will employ two functions that will optimize the data transfer throughput to and from a host computer. These two functions are packet filtering and packet compression. The Packet Filter table is used to determine which ProNET-80 data packets are desired by the host. The P80VME examines the packet identifier field (called a content label) in each received packet, and stores to on-board memory only those packets that are specified in the table. The Packet Compression table specifies the specific portions of the packets in the on-board P80VME memory to be transferred to the host. These two tables enable the P80VME to down load only the ProNET-80 packet information necessary for the application to perform its simulation task, thus reducing the I/O overhead.

Another requirement which was imposed by the Silicon Graphics Power Series workstation is the requirement for a DMA controller that can perform VMEbus Block Level Transfers. If BLTs are not used for data transfers between a VME device and workstation memory, a maximum data transfer rate of only 2 Mbytes/sec can be attained. By using BLTs, the transfer rate can be increased to approximately 9 Mbytes/sec in a workstation that utilizes SGI's IO2 VMEbus Adaptor. The IO2 is the bridge between the VMEbus and the workstation's main system bus, the MPlink Bus. This performance can be further improved in most Power Series workstations by replacing the IO2 with SGI's higher performance IO3 VMEbus adaptor. The IO3 supports data transfer rates of approximately 25 Mbytes/sec.

P80VME Hardware Architecture

The architecture of the P80VME consists of a RISC processor, static random access memory (SRAM), synchronous First-In-First-Out (FIFO) memories, a VMEbus interface chip set, and the Proteon p1582 ProNET-80 CTL interface (see Figure 4). All of the

components used in the P80VME design were chosen such that the optimum data throughput performance could be obtained.
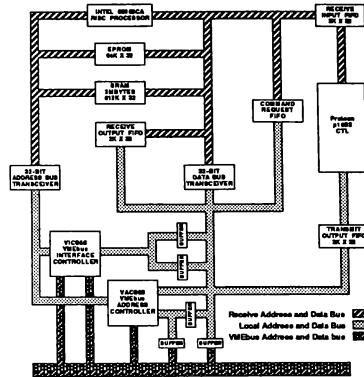

Figure 4. P80VME Block Diagram

In previous designs such as the P80GSI, the numerous memory management functions were performed using many high speed logic devices in order to obtain the performance required. This method of design made any memory management changes very difficult to implement and usually resulted in hours of circuit board rework and the development of new logic designs. This is not the case with the P80VME due to technology advancements in the area of RISC processors. The P80VME utilizes Intel's 80960CA 32-bit high performance embedded processor. The main responsibilities of the processor include packet and protocol management and high-speed data transfers. When processing a packet the processor determines if the protocol is correct, and based upon information contained in the protocol decides where to store the packet. Since these decisions are made by the processor it is very easy to accommodate changes in protocol and packet management by changing the embedded software. When using high-speed logic these changes are much more difficult. The processor is also responsible for the high-speed data transfers to and from different memory locations. The transfers are accomplished by using the burst capabilities of the processor. The ability to use a processor for these functions not only increases the flexibility of the design but also reduces the complexity of logic designs.

The memory system on the P80VME consists of 2 Mbytes of static random access memory (SRAM). This memory is made up of two 256k by 32 bit memory

modules with access times of 20 nanoseconds. These speeds provide the microprocessor with zero wait-state memory access.

Synchronous FIFO memories play a large role in the architecture of the P80VME. They are used for temporary storage of data coming off and going onto the network, host commands, and the storage of data to be transferred onto the VMEbus. These FIFOs have access speeds that support zero wait-state performance by the processor. The use of synchronous FIFOs enables the logic designer to use synchronous state machines instead of asynchronous designs that tend to be more difficult to design.

The VMEbus interface chip set used on the P80VME is the VIC068 and the VAC068 produced by Cypress Semiconductor. This chip set contains a DMA controller capable of block level transfers of up to 30+ Mbytes per second. This feature provides the most efficient means of moving data from the VMEbus into and out of a workstation's memory space.

The interface to the ProNET-80 network is provided by the Proteon p1582 CTL interface.[5] The CTL is a 6u VME card that supplies data on to and off of the ProNET-80 network at a rate of 80 Megabits per sec (Mbps). The CTL is mated with the P80VME to form a single 9u VME board (see Figure 5).

The use of these high performance components enables the P80VME to maintain data integrity, provide maximum throughput, and minimum I/O overhead to its host. By meeting these requirements the P80VME will be a major player in providing real-time capabilities for VMEbus computers in the ITDL.

Using The P80VME

A simulation application controls the P80VME through a software device driver located on the host computer. During a real-time simulation run, the workload of a host application associated with ProNET-80 I/O is minimized, as the only activities performed by the host is the writing of control commands to the P80VME. The 80960CA microprocessor and VIC068 VMEbus interface located on the P80VME perform the majority of the work by controlling and performing all data transfer activities between the P80VME and a host system.

During initialization the P80VME goes through its on-board self-test, checking on-board memory, FIFOs, and the CTL interface. Once this has been completed the processor is ready for commands from the application. The application should now set up the Packet Filter and

Packet Compression tables in the P80VME memory. Once these tables are set-up the P80VME is ready to receive simulation data and process it accordingly.
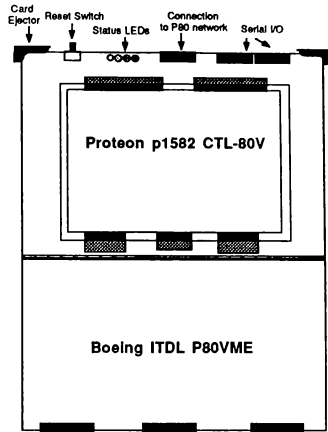

Figure 5. ProNET-80 VMEbus Interface

During a HIFEN2 simulation, data packets are transmitted in the broadcast mode. The p1582 CTL interface copies each of these packets off of the network, and P80VME logic then stores the data into a receive FIFO. When a complete packet is stored in the receive FIFO, the processor then transfers the data into the SRAM memory specified in the Packet Filter table. This task continues as long as there are complete packets in the receive FIFO. When a HIFEN2 synchronization packet is received the processor notifies the application that it is time to start a new frame by asserting a VME interrupt. The application then commands the P80VME to transfer the new frame of received data into its memory. The processor then uses the Packet Compression table to transfer only the portions of the packets desired into a FIFO memory. Next the processor sets up the DMA controller embedded in the VIC VMEbus interface to perform the transfer of the data into the host memory location that was specified by the host in the down load request command.

At an end of a simulation frame, the application is ready to transmit its vehicle state packet onto the ring. This process begins with the application building a transmit

packet in its own memory space. The application then writes a command to the P80VME containing the location of the packet and its size. The processor on the P80VME reads the command, and sets up the DMA controller in the VIC VMEbus interface which begins the transfer of the data into the transmit FIFO. Once complete, the processor writes the packet size to the transmit logic, and the packet is then transmitted onto the ProNET-80 network by the CTL interface.

Utilizing the features of Packet Filter and Packet Compression along with the on-board DMA controller, the P80VME optimizes the I/O throughput time so that the amount of frame time available for simulation processing is maximized.

Developing The P80VME

The development of the P80VME is being done by the engineering staff in the ITDL. The team is made up of a number of hardware engineers and software engineers each having specialized talents in different areas.

The hardware design is being done with the aid of Dazix CAD/CAE products. The schematics were drawn in a hierarchical structure using the ACE schematic capture package. The circuit board is a multi-layer two sided surface mount board that was defined using the BoardMaster package. Both ACE and BoardMaster are hosted on Intel 386 machines. The circuit board traces are being routed using Dazix's Star Router hosted on a Sun SPARC2 workstation. The programmable logic designs were done after the hardware architecture was defined and the schematics completed. The control lines were logically grouped into five Altera EPM5130 Erasable Programmable Logic Devices (EPLDs). The EPLDs are being designed using Altera's MAX+plus development tools hosted on an Intel 386 machine.

The embedded software to be executed by the P80VME's Intel 80960CA RISC microprocessor is being written primarily in the C programming language. In-line assembly and assembly functions are being used for time critical routines, and for setting up and initializing the microprocessor and its environment. Intel's 80960CA C compiler, assembler, and linker are being used, and are hosted on an Intel 386 system. The analysis and design of the embedded software utilized structured methods, and a PC based CASE tool was used for these activities. Initial test and debug of the embedded software is being done on Intel's EV80960CA Microprocessor Board that features an 80960CA microprocessor, SRAM memory, DRAM memory, and a serial port that is used to load software into its memory. Embedded software testing and debugging on

the P80VME will utilize the STEP Express 960CA In Circuit Emulator (ICE).

Test and debug of the P80VME will utilize a test bed consisting of a 9u VME chassis system, an Ironics SPARC based single board computer, and custom test bed software developed by the ITDL engineering staff. The SPARC board was chosen for two reasons. One, the board contains the VIC068 and VAC068 VMEbus interface chip set. This allowed initial testing of the chip set in the Silicon Graphics workstation to verify its compatibility and DMA capabilities. Two, a C compiler for the SPARC computer was already available in-house on a Sun SPARC2 station. The custom test bed software provides the capabilities to test the P80VME boards features. The software will act as an application requesting transmit and receive data transfers along with performing initialization and control functions. During testing, a Silicon Graphics workstation is not required until after the P80VME board has been fully tested on the test bed.

Conclusion

Real-time, distributed, multiple engagement, air combat simulations can impose unique and stringent requirements on local area networks. Commercially available networks are typically designed for campus or office network applications, and consequently they are often unsuitable for real-time flight simulation applications.

At the Integrated Technology Development Laboratories, Proteon's ProNET-80 local area network has been successfully adapted to provide the inter-participant communications for the HIFEN2 air combat simulation. Customized ProNET-80 interfaces that are optimized and tailored for real-time flight simulation networks have been developed for Gould and Silicon Graphics host computers. The custom interfaces have minimized host computer I/O processing overhead, which maximizes the amount of frame time available for simulation processing. These interfaces have also maximized the throughput from the network to a host computer, which has enabled HIFEN2 to support a greater number of simulated participants than its predecessor, HIFEN1.

The development of these custom ProNET-80 local area network interfaces for the HIFEN2 air combat simulation is just one example of the ongoing ITDL effort to adapt and apply new technologies for flight simulation applications.

References

[1]   Ronald G. Moore and Lee P. Emerson.
        "Evolution Of A Real-Time Air Combat
        Environment." AIAA Flight Simulation
        Technologies Conference, 1990, Publication
        90-3159-CP.

[2]   ProNET 10/80 Model p150x/p158x
        VMEbus/9u Network Interface with Fiber
        Optic Options Reference Manual. Document
        No. 42-040112-00. Revision A. May 1988

[3]   David M. Draffin and John S. Bacha.
        "Implementation of a Secure Multi-Project
        Laboratory Facility." AIAA Flight Simulation
        Technologies Conference, 1990, Publication
        90-3163-CP.

[4]   ProNET-80 Model p1680 Gould SelBUS
        Interface Installation and Programming Guide.
        Document No. 42-040068-00. Revision 1.0.
        January 1987

[5]   ProNET Model p1580 VME Local Network
        System Installation and Programming Guide.
        Document No. 42-040029-00. Revision 1.0.
        April 1986.

REAL-TIME INTEGRATION OF MAN-IN-THE-LOOP HYBRID SIMULATORS

Lisa Florio Girardi
Technical Writer
Calspan Advanced Technology Center
Buffalo, New York

Charles Edward Hagen
Principal Engineer
Calspan Advanced Technology Center
Buffalo, New York

James E. Hatem
Principal Engineer
Calspan Advanced Technology Center
Buffalo, New York

ABSTRACT

A USAF/Calspan study effort and integration demonstration were combined to provide the functional requirements and demonstrate the feasibility of a real-time integration of the REDCAP facility with other electronic warfare simulation facilities. The study was performed in support of the Office of the Secretary of Defense initiatives to avoid duplication of electronic combat (EC) test community efforts and enhance capabilities by increasing test-asset utilization. Its results defined concepts for linkages that could provide real-time, man- in-the-loop feedback for defensive-system cueing and offensive- system integration critical to advanced airborne weapons systems using computer-aided situation-awareness techniques. The study also generated detailed requirements for implementing a real-time integration of upgraded REDCAP with other ECT&E assets; specifically, the upgraded Air Combat Environment Test and Evaluation Facility (ACETEF). A feasibility demonstration of such a link was successfully conducted on 27 September 1990. Successful linkage has also been achieved between
REDCAP, the Electromagnetic Test Environment (EMTE) facility and the Air Force Electronic Warfare Evaluation Simulator (AFEWES). Interfaces with the Electronic Combat Integrated Test Facility (ECITF) at the Air Force Flight Test Center (AFFTC) and other EC facilities are being investigated.

INTRODUCTION

The purpose of this paper is to describe real-time integration (including feasibility, requirements, and benefits) of the REDCAP facility with other electronic combat simulators. REDCAP linkage with other electronic combat test and evaluation (ECT&E) facilities will provide required real-time man-in-the- loop C^3 capabilities to other electronic combat facilities that have terminal threat capabilities. Such linkages have been recently demonstrated with the EMTE Range and AFEWES, as well as ACETEF. Other facilities that potentially could benefit from real-time integration include ECITF, Pre-flight Integration of Munitions and Electronic Systems (PRIMES), Guided Weapons Evaluation Facility (GWEF), and other EC test assets. We will discuss here the REDCAP facility's

recent integration demonstrations and fully integrated facility plans.

The recent studies and demonstrations have indicated that permanent linkages between REDCAP and other real-time facilities would provide major enhancements to urgently required test capabilities at little additional cost. Such linkages also offer an opportunity to combine the strengths of multiple facilities and significantly improve DoD's EC test capability by providing fully interactive, cost-effective, high-fidelity EC simulations. The unique capabilities of individual facilities would complement and supplement each other in areas where large effort-duplication costs can be avoided. Linkages would allow, for the first time, unique system testing against a modern, integrated air-defense system with realistic system loadings. The links, themselves, would pose little technical risk and would provide unique near- term capabilities to test the interaction of advanced systems against an integrated, modern air-defense system--land-, sea-, or air-based.

The links are achieved with off-the-shelf commercially available equipment to produce a low-risk implementation. The facilities require communications equipment to hook to the T-1 telephone lines, cryptological equipment (KG- 94s), and intelligent multiplexers. A MIL-STD-1553B remote bus interface is needed to hook up systems under test.

Each of three major linked facilities that have already conducted demonstrations (REDCAP, EMTE, and ACETEF) will be briefly described below. This will be followed by a description of the current integration efforts.

FACILITY DESCRIPTIONS
REDCAP

REDCAP is a major Air Force electronic combat simulation facility that is used to evaluate actual U.S. electronic combat equipment,
concepts, and tactics against foreign integrated air- defense systems. It is located at Calspan Corporation's Advanced Technology Center (ATC) in Buffalo, New York, and has been developed and refined over a 25-year period. REDCAP is a laboratory hybrid test resource that combines all the elements of an integrated air defense system (IADS) with real-world signal densities in real time.

While principally a development test and evaluation (DT&E) facility supporting technical development, it has also been used for flight test planning and data extension for operational test and evaluation (OT&E). By using a hybrid simulation concept, which includes actual RF equipment and man-in-the-loop techniques, the system provides a complement to both flight test and computer simulations. However, a current limitation to REDCAP is that it represents man-in-the-loop Blue penetrator reactions with simple computer simulations that provide an essentially "canned" Blue response.

REDCAP is used to simulate the threat air-defense system, starting with the radars that detect penetrators and continuing up through the echelons of battle management and various weapons- control points. At key points, simulated weaponry is directed against the vehicles penetrating the air-defense system.

Simulation Capabilities

The overall simulation consists of hundreds of simulated radars, a subset of which contain actual radar receiver simulations at real radar frequencies with real processors. The outputs, shown on radar displays, are manned by experienced operators who have been trained in foreign air-defense procedures and doctrines. The many other radars in the network use computer simulations based on measurements of the performance of actual operators at the real displays (or from other facilities), but these simulated radars drive environment generators to produce the proper RF

(radio frequency) environment for systems under test, which might typically be the receiver-processors and transmitter-exciter subsystems of a developmental EC system for IADS network jamming.

The operators and simulations detect, track, and distribute target information to the IADS. At the IADS posts, the operators filter information from the large number of reports and decide which targets are threats, which are friendly, and what defensive action will be taken. Typically, the decisions result in interceptors or surface-to-air missiles (SAMs) being assigned. If appropriate, REDCAP simulations of the interceptor aircraft-- manned by real pilots--then attempt to make the intercept.

As briefly mentioned earlier, a typical REDCAP exercise might require testing new U.S. electronic combat equipment that attempts to exploit potential weaknesses in an air defense system--for example, an aircraft using deceptive radar jamming techniques so that it can fly through an area without being correctly identified and engaged. To support this type of test, REDCAP would run a series of scenarios with vehicles penetrating the area with various jamming techniques, flight paths, and altitudes. From the data gathered in these scenarios, Air Force and Calspan personnel would evaluate how effective the IADS is in tracking the jamming vehicle, the probability and frequency of intercept or SAM engagement, and the time it took for defensive reactions to occur.

REDCAP Test Environment

The REDCAP test environment consists of several elements-- detection systems, C^3 (command, control, communications) and weapons control communications, radar and communication environment simulators, and test control and data reduction. REDCAP provides, at RF, air-to-air,

air-to-ground, and ground-to- air data links; identification, friend or foe (IFF); voice links; and large-scale, ground-based computer simulations.

One of the more important detection systems in REDCAP is the Soviet Union Airborne Warning and Control System (SUAWACS) simulator. This complex, real-time simulator permits evaluation of tactics, ECM techniques, and equipment targeted at the SUAWACS threat. The SUAWACS simulator has been used extensively in testing ECM systems. As in all other REDCAP tests involving penetrator RF receivers or transmitters, the ECM hardware is installed in a screen room that is part of the REDCAP real-time RF simulation facility to eliminate any coupling not involving the correct propagation paths between antenna ports.

EMTE

The Air Force Development Test Center maintains an electromagnetic test environment (EMTE) to support developmental and operational agencies in the test and evaluation of electronic combat devices, components, systems, and techniques against simulated hostile defense systems. The EMTE complex consists of numerous radars that operate in different frequency bands and modes to provide a flexible test capability. It comprises 22 test sites, 46 threat simulators, and 26 support facilities. The complex is complemented by electronic countermeasure (ECM) ground monitoring facilities that receive, record, and display RF spectral and time domain data of received signals over a broad spectrum. EMTE is also capable of providing dynamic aircraft radar cross-section data, antenna measurement data of airborne devices, and has video playback capability for detailed post- mission analyses.

EMTE is updated continuously to provide current threat simulations

and enhanced instrumentation capabilities. Users include the 3246th Test Wing, the Tactical Air Warfare Center, operational units of the Tactical and Strategic Air Commands, and others, all of whom will greatly benefit from EMTE's enhanced capabilities through the REDCAP link.

ACETEF

ACETEF, located at the Naval Air Test Center (NATC) in Patuxent River, Maryland, is a multi-spectral, man-in-the-loop, hardware-in-the-loop test facility integrated around an aircraft- sized anechoic chamber. ACETEF comprises a Closed-loop Threat Simulator, the Manned Flight Simulator, and the Electronic Warfare Integrated Systems Laboratory. The facility is being upgraded to include a Communication, Navigation, and Identification Laboratory for both friendly and threat systems, an Operations and Control Center, an Offensive Sensor Laboratory, and both Red and Blue C2 centers.

ACETEF supplements flight testing efforts with realistic ground tests. Its function is to increase the scope of testing that can be realistically performed within today's constraints by providing a logical approach to generating flight test scenarios. Data generated by flight tests will be used to validate the laboratory tests performed.

Much of the new electronic countermeasures equipment is not as "independent" as a pod suspended under a wing. Such new equipment is often controlled by the same computer that controls many of the aircraft's other functions. It is essential to test highly integrated systems to determine real-world reactivity. It is not possible to bring an aircraft into REDCAP, but it is possible to put one in the anechoic chamber at ACETEF, where REDCAP can provide the high-fidelity C2 inputs to ACETEF subsystems and environment generation.

This simultaneously mitigates an ACETEF limitation of Red C^3 reactions that are essentially "canned" routines in a digital computer simulation, and the previously mentioned REDCAP limitations with regard to the reactivity of highly integrated Blue systems.

REDCAP/EMTE Integration Efforts

REDCAP has also successfully linked with the Electromagnetic Test Environment (EMTE) facility at Eglin AFB in Florida. Via EMTE, that link included AFEWES in Ft. Worth, Texas, as well. The REDCAP/EMTE/AFEWES effort currently included several demonstration runs earlier this year. A link with ECITF at Edwards AFB is also being discussed.
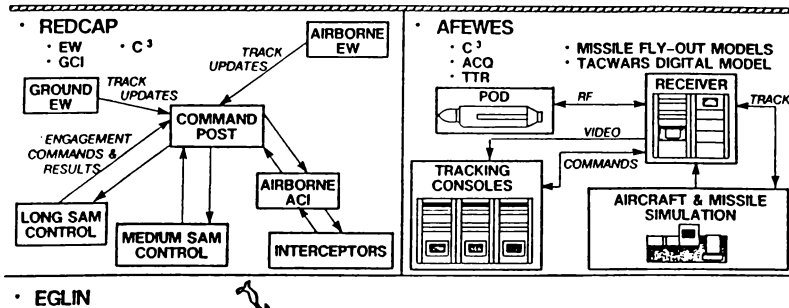
EMTE is an electronic combat range test environment. Through the linkage, REDCAP provided real-time C^3 simulation for EMTE's open-air range. Eglin has actual radar hardware on the range, but very limited threat command and control capabilities. REDCAP provided that, performing such functions as realistic cueing of test operators at EMTE for pointing SAMs at targets and other required procedures, including countermeasures against C^3, for the scenarios tested.The scope of the integrated test facilities is shown in Figure 1.

For the expanded link between REDCAP, EMTE, and AFEWES, the data supplied by REDCAP to AFEWES will be filtered through EMTE. The link is expected to add an early look at system capabilities and increase the probability of mission success by allowing near- real-time changes of system parameters, modes, and flight profiles. The integration of laboratory simulations with an actual airborne or ground test will potentially improve test realism, the fidelity of air-defense simulations, and the validity of test results.

The objectives of the REDCAP/EMTE/AFEWES link include demonstrations of the methodologies

Figure 1

INTEGRATED TEST FACILITIES
SCOPE

- REDCAP
  - EW · C³
  - GCI

  AIRBORNE EW
  TRACK UPDATES

  GROUND EW TRACK UPDATES

  ·ENGAGEMENT COMMANDS & RESULTS

  COMMAND POST

  AIRBORNE ACI

  LONG SAM CONTROL
  MEDIUM SAM CONTROL
  INTERCEPTORS

- AFEWES
  - C³
  - ACQ
  - TTR

  · MISSILE FLY-OUT MODELS
  · TACWARS DIGITAL MODEL

  POD RF RECEIVER TRACK
  VIDEO
  TRACKING CONSOLES COMMANDS
  AIRCRAFT & MISSILE SIMULATION

- EGLIN

  A-11
  A-12 A-6 A-3
  A-17A

needed to measure and display in real-time: 1) the capability of support jamming systems to degrade terminal threat engagement opportunities; 2)the capability of self-protection ECM systems to counter terminal threats as netted in an air-defense system; and 3) the synergistic affects of simultaneous employment of support jamming and self-protection jamming to air-defense system performance degradation.

For the initial three-way demonstration, REDCAP provided a simulated support jammer platform and a SUAWACS aircraft, both flying scripted profiles. Two hostile airborne interceptors were also simulated to support the SUAWACS. EMTE provided two F-16 A/B aircraft with one as a back-up for the mission. The aircraft had an operational radar warning receiver (RWR) and an ECM pod on board. A transponder for accurate tracking by ECM instrumentation radars was installed on the F-16s. AFEWES

provided a Flight Simulator Lab (FSL) F-16 simulated aircraft with JETS ECM jamming to fly as wingman support to the EMTE aircraft. Two F-16 scripted target aircraft were also simulated for the demonstration.

For the first objective listed above, the methodology used was evaluation of the performance of baseline terminal threat assignment times and subsequent engagement times for each of the four penetrating aircraft and threat laydowns during one run with the help of support jamming. The purpose was to evaluate the contribution of support jamming to mission success when simultaneously employed with self-protection jamming.

The purpose of objective two was to demonstrate the contribution of self-protection jamming to individual threat performance degradation in this linked environment. This was achieved by activating self-protection ECM using the jammer pod on board the penetrating EMTE F-16 and JETS

simulated jamming. No support jamming by the simulated EF-111 was provided for this run.

The methodology for objective three was the same as for above, but with simulated support jamming ECM activated to provide both self-protection and support jamming during the run. Its purpose was to show use of integrated test facilities to extend the scope of EC test and evaluation capabilities. Figure 2 illustrates the demonstration scenario.

REDCAP facility can operate in an integrated fashion with manned friendly assets in the ACETEF facility. This capability has the potential to enhance testing at many locations.

The demonstration included both airborne and land-based RF man-in-the-loop early warning systems, manned threat command and control and interceptors, RF signal generation, manned friendly interceptors, and joint test control all linked via secure communications. Figure 3

Figure 2

REAL-TIME DEMONSTRATION SCENARIO (LAYDOWN)

- F-16 SURVIVABILITY
- TEST ASSET RECONFIGURABILITY
- EC ASSET SYNERGISTIC EFFECTIVENESS
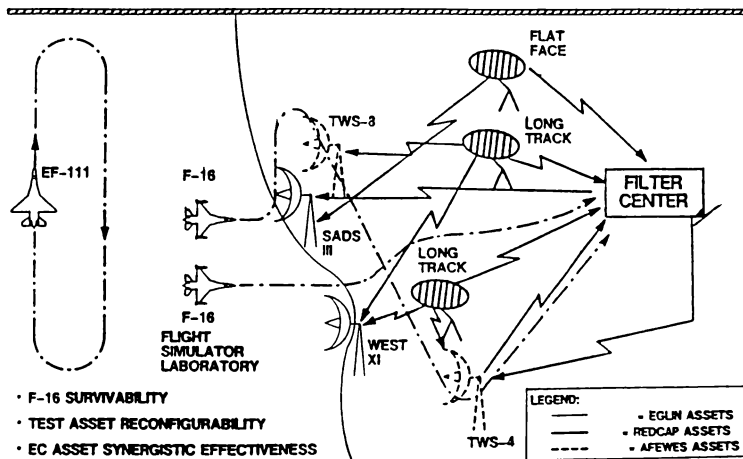
REDCAP/ACETEF INTEGRATION EFFORTS

The interaction of ACETEF's high-fidelity Blue reactive environments with the high-fidelity Red active capability of REDCAP provided for full testing capabilities for new, highly integrated smart aircraft. The real-time effort, conducted in the fall of 1990, demonstrated that manned threat command and control elements of the

illustrates the scenario used in the demonstration.

The purpose of the REDCAP/ACETEF demonstration was three-fold. First, the Air Force hoped to demonstrate the feasibility and potential utility of a real-time interconnection between REDCAP and ACETEF (and, therefore, the feasibility of linkages between REDCAP and other real-time

Figure 3
REDCAP/NATC NEAR TERM DEMONSTRATION
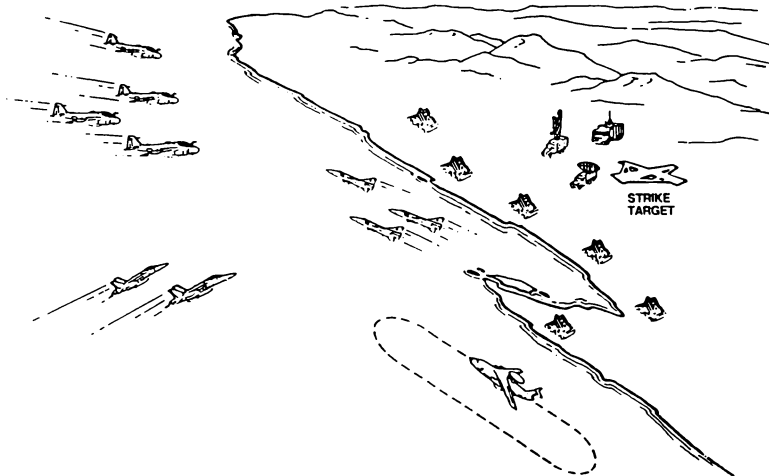
facilities, such as EMTE or ECITF). Second, it was necessary to demonstrate REDCAP's ability to provide appropriate stimuli to a remote (ACETEF) threat environment. And third, to demonstrate remote (ACETEF) penetrator support and terminal threat engagement feedback to REDCAP.

The linked REDCAP/ACETEF facilities would allow for more complete testing of aircraft systems in a secure, controlled environment than previously available. The combined capabilities of REDCAP and ACETEF provide a synergy that allows for testing of advanced technology systems. This synergy combines the best capabilities of those two unique national EC test assets. A complete weapons-system test with RF threat systems will be available for EC simulations. Those threat systems include air defense command and control and terminal threats with man-in-the-loop interaction, complete aircraft-in-the-loop integration, high- fidelity visual cueing,

offensive sensors integration, and complete interoperability capabilities including high-fidelity RF background. Additionally, new capabilities will be available (such as an integrated Naval Air Defense System (INADS) and potential Naval surveillance radar simulations), providing for additional usage of government assets.

Linkage-generated test conditions, previously not producible anywhere but in flight tests, will allow facilities to interact with each other to help determine mission success.

Testing of highly integrated weapons systems or interactions between weapons systems would now be possible. Tests such as SUAWACS or RADAR SOJ, BMC^3 Link Denial, Situational Awareness, Data Link Deception, and Drone Jamming could be more effectively conducted simultaneously with installed performance measurements in combinations or per mutation to test

the overall EC effort.

ACETEF and REDCAP functions, listed below, show the synergism between the two. REDCAP provides man-in-the-loop Red surveillance C^3 with data provided from Red sensors, Red communications, and the Red environment to stimulate ACETEF's environment generator capabilities to feed the aircraft anechoic facility. The REDCAP high echelon SAM C^3 provided for proper control of the terminal Red SAMs located at ACETEF while Red pilot stations at REDCAP attempted to engage the Blue pilots at ACETEF. All these actions allowed ACETEF to provide Blue surveillance C^3 for supporting man-in-the-loop decision making and fully integrated system testing.

The scenario used for the demonstration was limited to an over-the-beach strike using Naval assets as pictured below. It was shown that the integration of two geographically separate hybrid simulators is possible and would greatly benefit the EC community. Some of the enhanced capabilities highlighted in the demonstration include highly reactive manned Red and Blue aircraft, both Red and Blue C^3, offensive and defensive high fidelity aircraft-in- the-loop stimulation, human factors analysis, tactics evaluation, complete end-to-

end system testing, and a mutually expanded threat domain.

This demonstration has proved that other facilities with similar real-time capabilities (such as ECITF, EMTE, and AFEWES) could also successfully integrate with REDCAP's assets, thereby increasing each facility's utility and decreasing redundancies in EC test assets.

While the link was being demonstrated, Calspan engineers began investigating the requirements of a long-term link. The approach taken was to: define the potential test scenarios, determine the potential threat, determine which tests or subsets of tests can be supported by existing or planned assets at the various facilities, determine what data need to be transferred to tie these assets together, determine the rate at which these data need to be transferred, and, finally, to determine what equipment is required to support the joint test capabilities. Once these parameters were established, a determination was made on the relative magnitude of complexity of each. Additional areas where improved test capabilities can be obtained (such as adding Naval surveillance radar simulations) were also identified. Data required for transfer and functional linkage are depicted in Figures 4 and 5.
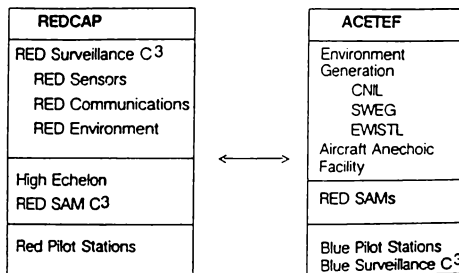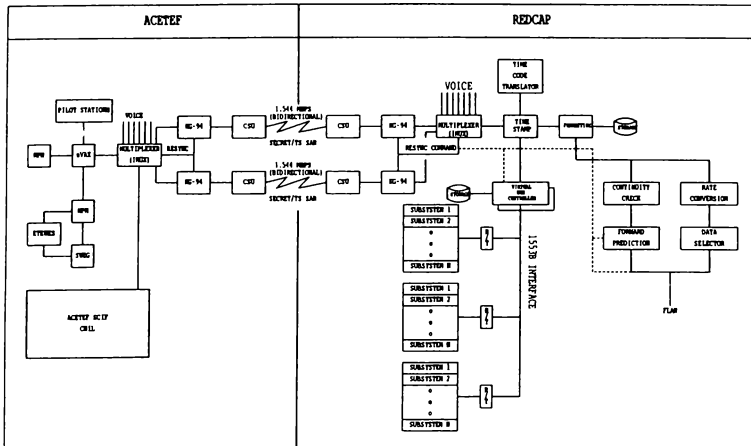
Figure 4

ACETEF/REDCAP FUNCTIONS

REDCAP	ACETEF
RED Surveillance C^3 RED Sensors RED Communications RED Environment	Environment Generation CNIL SWEG EWISTL Aircraft Anechoic Facility
High Echelon RED SAM C3	RED SAMs
Red Pilot Stations	Blue Pilot Stations Blue Surveillance C^3

Conclusions

The successful integration of REDCAP with other real-time facilities, including ACETEF, EMTE, and, AFEWES, proves that it is not only possible, but preferable, to combine the assets of the individual EC facilities to enhance the test capabilities of each and provide more comprehensive, realistic simulations of potential real-world encounters. In doing so, redundancy of test efforts by the various branches of the military would be virtually alleviated, saving millions of tax payers' dollars. Figure 6 lists test capabilities enhanced by integration.

With the demonstration links successfully completed, REDCAP engineers are working toward the establishment of long-term linkages with these facilities so that the goals stated above can be achieved on a broader scale.

Figure 6
ENHANCED INTEGRATED CAPABILITIES
AFDTC Facilities (REDCAP, AFEWES, EMTE, PRIMES, & GWEF)

- REACTIVE AIRCRAFT
  High Fidelity Manned Red Aircraft
  High Fidelity Manned Blue Aircraft
  Multiple Lower Fidelity Manned
    Aircraft

- COMMAND, CONTROL, AND
  COMMUNICATIONS
  Red C^3 and Blue C^3

- HIGH FIDELITY AIRCRAFT SIMULATION
  Both Offensive and Defensive

- HUMAN FACTORS
  Complete Manned Mission Analysis

- TACTICS EVALUATION

- END-TO-END TESTING
  Mission Evaluation
  Terminal Threats

- THREAT DOMAIN EXPANSION
  Naval Threats
  INADS

Plus Integrated Facilities: Upgraded ACETEF & ECITF

THE ORIENTATION REPRESENTATION IN THE DRAFT MILITARY STANDARD FOR DISTRIBUTED INTERACTIVE SIMULATION

Brian Goldiez* and Kuo-Chi Lin**
University of Central Florida
Orlando, FL 32826

Abstract

The use of Euler angles or Quaternions to represent entity (i.e., vehicle) orientation in the draft military standard for Distributed Interactive Simulation has drawn fierce debate. The debate involves how to best represent and transmit vehicle orientation in a network of simulators. This paper analyzes these two representations for vehicle orientation by considering three pertinent factors; network traffic, efficiency of computation, and utilization by simulator manufacturers. Network traffic is analyzed by considering the amount of data generated by each representation. Efficiency of computation considers sample execution speeds of each method using a FORTRAN implementation. Utilization by simulation manufacturers considers the results of a survey conducted by the Institute for Simulation and Training at the University of Central Florida. The conclusions drawn from these analyses show that Euler angles are the better method of communicating vehicle orientation in networked simulators.

1. Introduction

The use of Euler angles or Quaternions to represent entity (i.e., vehicle) orientation in the draft military standard for Distributed Interactive Simulation has drawn fierce debate.[1] The debate involves how to best represent and transmit vehicle orientation in a network of simulators. In the computer simulation of the motion of a vehicle, there are three methods for representing the orientation of the vehicle. The first method uses Euler angles, which consist of an ordered set of three successive rotation angles to represent the orientation of the vehicle. A second method involves the use Quaternions, which is a four-parameter system first developed by Euler in 1776. The four parameters of the Quaternion correspond to the unit vector of the rotation axis and the scalar rotation angle. A third method for representing the orientation involves the use of a rotation matrix, which consists of nine direction cosines of the body axis relative to the world axis to represent the orientation Each method has been used successfully in the past in different areas. However, since the rotation matrix is not efficient in either kinematics computation or network traffic, it is usually calculated from Euler angles or Quaternions. The formulas are listed in Appendix A. The discussion in this paper will focus only on the comparison between Euler angles and Quaternions.

The origin of Euler angles and Quaternions can be found in the literature. A set of Euler angles or Quaternions are useful to transform from the body or vehicle fixed frame of reference to an inertial frame of reference. The importance is obvious for navigation purposes, simulated impact with the terrain, and visual system representation. The implications for networking are less obvious. Obviously a network of simulators must convey their respective locations for the reasons noted above. However, one must maintain maximum computational efficiency and minimize network bandwidth. These constraints are often contradictory.

Each method of transformation has its limitations. Euler angles are computed using transcendental functions which can be computationally expensive. Euler angles also have a singularity point. In the case of aircraft conventions, the order of rotation of yaw, pitch, and roll yields a singularity at a pitch angle of $\pi/2$. Quaternions are limited due to their lack of direct correlation to observable and measurable parameters.

In a distributed interactive simulation system, any one of the simulators not only calculates its own orientation but also needs to know the orientation of other vehicles in the same battlefield for the purpose of graphics display, weapons impact, sensors, etc. Hence it is required that the orientation information be periodically sent to a simulator by each of the other simulators on the network.[2] Which of the above mentioned methods should be chosen as a standard for the orientation representation must be carefully evaluated. The choice of representing entity orientation for simulation networks should be based on three factors: 1. computational efficiency at the sending and receiving ends, 2. network traffic, and 3. utilization of the values within a simulator. These factors will be discussed in the following sections.

* Director, Research & Development, Institute for Simulation and Training, member AIAA

** Assistant Professor, Department of Mechanical and Aerospace Engineering/Institute for Simulation and Training, member AIAA

2. Background

2.1 Kinematics of the Vehicles

A vehicle in general three-dimensional motion has three degrees of freedom in rotation. Thus it needs three differential equations usually the Euler equations, to govern the rotation motion. The integration of these three differential equations leads to the three components of angular velocity in the body axis system. The kinematic equations are then used to convert the angular velocity into the system which is chosen to represent the orientation of the vehicle. The direction cosines are seldom used as the variables in the kinematic equations because of their complexity. Therefore, they will not be considered. On the other hand, Euler angles and Quaternions are almost equally popular. The advantage of using Euler angles is that it is easy to understand the physical interpretation. Most engineers can picture the orientation of the vehicle by knowing Euler angles. The disadvantage of Euler angles is the singularity problem. When the pitch angle of the vehicle is at 90°, the roll and yaw motion can not be defined, which is often called gimbal lock position. The numerical integration of the kinematic equations also has difficulty for a pitch angle close to 90°. This is the major advantage of using Quaternions which do not have above the mentioned singularity problem. Minimal transcendental function for Quaternions appear to lend themselves to solution on digital computers easier than Euler angles. Quaternions, on the other hand are more difficult to read and interpret. The disadvantage of Quaternions is their use of four numbers hence their need of four differential equations which is one more than the degrees of freedom of the vehicle. An algebraic constraint equation also needs to be imposed on the kinematic equations which increases the difficulty of numerical integration.

2.2 Dynamic Equations of Motion

The linear equations of motion can be either in the fixed or in the body-axis coordinates.[3] If the body-axis coordinate is used, the linear equations of motion are

$$m(\dot{U} - VR + WQ) = F_x \qquad (1)$$
$$m(\dot{V} - WP + UR) = F_y \qquad (2)$$
$$m(\dot{W} - UQ + VP) = F_z \qquad (3)$$

where U, V, W are the velocity components in body-axis coordinate, P, Q, R are the angular velocity components in body-axis coordinate, and F_x, F_y, F_z are the total applied force components in body-axis coordinate.

The rotational equations of motion are usually in the body-axis

coordinate as (assume xz-plane symmetry)

$$I_{xx}\dot{P} + I_{xx}(\dot{R} + PQ) + (I_{zz} - I_{yy})QR = M_x \qquad (4)$$
$$I_{yy}\dot{Q} + (I_{xx} - I_{zz})PR + I_{xx}(R^2 - P^2) = M_y \qquad (5)$$
$$I_{zz}\dot{R} + I_{xx}(\dot{P} - QR) + (I_{yy} - I_{xx})PQ = M_z \qquad (6)$$

where M_x, M_y, M_z are the total applied moment components in body-axis coordinate. These equations are solved for P, Q, and R. When integrated, the equations for P, Q, and R can be used to compute Euler angles and Quaternions

2.3 Euler Angles

The equations of Euler angular rates are

$$\dot{\psi} = (Q \sin\phi + R \cos\phi)/\cos\theta \qquad (7)$$
$$\dot{\theta} = Q \sin\phi + R \cos\phi \qquad (8)$$
$$\dot{\phi} = P + \psi \sin\theta \qquad (9)$$

The velocity components in fixed coordinate are computed by the following matrix multiplication

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix} = [\mathbb{R}]^T \begin{bmatrix} U \\ V \\ W \end{bmatrix} \qquad (10)$$

where $[\mathbb{R}]$ is the rotation matrix. The expression of the rotation matrix in terms of Euler angles can be found in Appendix A.

2.4 Quaternions

The equations of Quaternions rates are

$$\dot{e}_1 = \frac{1}{2}(-Pe_4 - Qe_3 - Re_2) \qquad (11)$$
$$\dot{e}_2 = \frac{1}{2}(-Pe_3 + Qe_4 + Re_1) \qquad (12)$$
$$\dot{e}_3 = \frac{1}{2}(Pe_2 + Qe_1 - Re_4) \qquad (13)$$
$$\dot{e}_4 = \frac{1}{2}(Pe_1 - Qe_2 + Re_3) \qquad (14)$$

The values of Quaternions are constrained by

$$e_1^2 + e_2^2 + e_3^2 + e_4^2 = 1 \qquad (15)$$

To compensate the violation of constraint, a term is added to

each of the differential equations as

$$\dot{e}_1 = \frac{1}{2}(-Pe_4 - Qe_3 - Re_2) - ke_1\Phi \tag{16}$$

$$\dot{e}_2 = \frac{1}{2}(-Pe_3 + Qe_4 + Re_1) - ke_2\Phi \tag{17}$$

$$\dot{e}_3 = \frac{1}{2}(Pe_2 + Qe_1 - Re_4) - ke_3\Phi \tag{18}$$

$$\dot{e}_4 = \frac{1}{2}(Pe_1 - Qe_2 + Re_3) - ke_4\Phi \tag{19}$$

where k is a constant and

$$\Phi = e_1^2 + e_2^2 + e_3^2 + e_4^2 - 1 \tag{20}$$

is the amount of constraint violation. If the constraint is satisfied,

$$\Phi = 0 \tag{21}$$

As with Euler angles the velocity components in fixed coordinates are

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix} = [R]^T \begin{bmatrix} U \\ V \\ W \end{bmatrix} \tag{22}$$

Here, again, $[R]$ is the rotation matrix. The expression of the rotation matrix in terms of Quaternions can be found in Appendix A.

3. Comparison

Quaternions and Euler angles have distinct advantages. Euler angles are commonly used in simulators as a driver for various systems and instruments which rely on an angular reference from some datum. Euler angles are also represented by three values as opposed to four values for Quaternions. Quaternions, as implied above, have no singularities. ' In addition, Quaternions can be determined directly from the equations of dynamic motion, without using time consuming transcendental functions.

3.1 Network Traffic

The question of sending Euler angles or Quaternions over a network of simulators is evaluated directly with respect to network traffic and dead reckoning algorithms. Fidelity requirements play an indirect part in the trade-off. From a network traffic point of view Euler Angles make more sense than Quaternions to send between entities in a distributed simulation. Euler angles generate three values per update, while Quaternions generate four values per update.

3.2 Computational Efficiency

Various techniques can be devised to work around the singularity for Euler angles at a pitch attitude of $\pi/2$. These include truncation of values as $\pi/2$ is approached to restrict division by zero. Angular measure can then be inverted to skip over the singularity. The difference between the cosine of $89.99°$ or $90.01°$ and $90°$ is 1.745×10^{-4}. The inverse of this value is approximately $5,700$. The inverse, or secant, is used in the computation of yaw rate. These values are well within the dynamic range of real-time computers for accurate computations using floating point or fixed point ($2^{16} = 65,536$ for 16 bit word) operations. Therefore an accuracy of $.02°$ can be easily achieved using microprocessor technology. In addition, the current SIMNET implementation uses a $.3°$ threshold for a PDU update. Accuracies which are an order of magnitude greater in accuracy than current implementations, therefore, are easily implemented. These type of accuracies, if used judiciously can avoid the singularity problem. Quaternions generate 33% more traffic than Euler angles due to four Quaternion values being required to transform coordinated vs. three Euler angles.

Quaternions are constrained by the fact that the sum of the square of the Quaternion values must equal one. This constraint is normally not satisfied in digital computers due to round-off and truncation errors. This constraint should be forced through artificial means to avoid positional drift over long time periods. This constraint should also be forced in fast moving vehicles to preclude discontinuities in positional updates.

Dead reckoning models can indirectly impact whether Euler angles or Quaternions are used. High order dead reckoning or extrapolation models require first and possibly higher order derivatives in order to determine position. First order derivatives are a natural outcome from the calculation of Quaternions. These values are then integrated to determine the Quaternion values. Euler angles are then computed using transcendental functional relationships from the Quaternions. Euler angle rates are not a normal output from a simulation mathematical model which computes the Euler angles from Quaternions. Therefore, higher order dead reckoning models can require additional computations of Euler angle rates, which can then be used in higher order dead reckoning models. If Euler angles are computed directly from the dynamic equations, then their rates are also immediately available as shown in Eqs. (7), (8), and (9). Higher fidelity simulations which require higher order dead reckoning models can use either

method and avoid additional computations in the generation of extrapolated vehicle position. If the first order dead reckoning is used to update the orientation, the rate or orientation variables are also needed in the network. In this case, Euler angle rates will be a problem for the pitch angle near singularity.

Simulators on a network will be of various types and complexities. It is, therefore, difficult to hypothesize a priori a need for higher order dead reckoning models. This paper recommends zero order components be used for vehicle orientation across the network. Higher order terms can then be computed using various numerical differentiation routines on the receiving node. This approach only burdens the node which requires the data to produce the data.

In order to better assess the relative computational efficiency of computing Euler angles and Quaternions, a simple test was developed. Separate FORTRAN programs were written to determine Euler angles and Quaternions. Simple computational efficiencies were taken in the Euler angle program to minimize the number of transcendental function computations. These included computing $\sin\theta, \cos\theta, \sin\phi, \cos\phi$, $\sin\psi$, and $\cos\psi$ once. The values were then stored as real variables, avoiding duplicative transcendental computations in the rotation matrix. Other efficiencies to minimize duplicative multiplications and divisions were not made. We assume such efficiencies would equally benefit both implementations. Truncation of Euler angles at 89.99 and angular reversal to yield the equivalent of 90.01 is also assumed to have minimal computational impact on Euler angle computations.

The separate FORTRAN implementation of Euler angles and Quaternions yielded differences in execution time. Specifically, on a PC/386, 5000 Euler angle computations were executed in 20.38 seconds while 5000 executions of Quaternions (with Euler angles computer from Quaternions for instrument drivers) were computed in 27.63 seconds. Software listings are included in Appendix B for both implementations. Differences in computational efficiency is felt to be negligible between Quaternions and Euler angles when gimbal lock considerations are included in each Euler angle computations simple limiters. The *30%* improvement in Euler angle computations can be used to accommodate anti-gimbal lock algorithms.

3.3 Survey Results

It is important to note that simulators for which Institute of Simulation and Training has access to math models or source code all indicate an area where Euler angles are computed. The use of these values varies. Some simulators use the values for instruments and airframe systems simulations. Some simulators use the Euler angles for computation of spacial orientation. Some simulators use Quaternions for computation of spacial orientation. Almost all higher fidelity (i.e., those using 6 degree of freedom models and dynamic model update rates of 30Hz) simulations use Quaternions to represent spacial orientation because of short update periods and simplicity (i.e., no need to work around the gimbal lock problem). In order to get an indication of where and how Euler angles and Quaternions are used, the Institute for Simulation and Training at the University of Central Florida created and conducted a survey from industry. The following questions were asked.

AVAILABILITY

I.1. Indicate if both Euler angles and Quaternions are available for network transmission from your simulator(s). Indicate if your simulator uses a mix of Euler angles and Quaternions to represent vehicle orientation.

I.2. Indicate if both Euler angles and Quaternions are available for network transmission from your simulators. Indicate if your simulator uses Quaternions for ownship orientation and Euler angles for other purposes.

II. Indicate if only Euler angles are available for network transmission in your simulator(s).

III. Indicate if only Quaternions are available for network transmission in your simulator(s).

IV. Indicate if your simulator(s) currently are unable to transmit Euler angles or Quaternions.

INPUT TO GRAPHIC DISPLAYS

I. Indicate if your simulator's graphics displays accept Euler angles easiest.

II. Indicate if your simulator's graphics displays accept the Rotation Matrix (Direction Cosines) easiest.

III. Indicate if your simulator's graphics displays accept Quaternions easiest.

Approximately 200 questionnaires were sent our to organizations who attended the Distributed Interactive Simulation Standards workshop hosted by the Institute for Simulation and Training in August 1990. 47 replies have been received by IST as of April 10, 1991. The results are:

Availability	Organizations answering affirmative	Number of simulator types affected[†]
I. 1	9	19
I. 2	5	17
II.	12	34
III.	2	19
IV.	2	2

Input to Graphics Displays	Organizations answering affirmative	Number of simulator types affected[†]
I.	20	47
II.	11	46
III.	2	6

One can therefore conclude, if this author's informal survey is accurate, that Euler angles are computed more often in simulators than Quaternions; but a significant number of simulators compute both Euler angles and Quaternions.

Processors for the graphic display systems used on the simulator accept either Euler angles or the rotation matrix as input. If the orientation variables from network match the graphic display's input style, no conversion process is needed. Otherwise, the orientation variables need to be converted to the proper representation. For example, if the simulator accepts rotation matrix as graphic display input and the orientation variables coming from network are either Euler angles or Quaternions, they need to be converted to the rotation matrix. In this case it is more efficient to convert Quaternions to the rotation matrix because there is no trigonometric function involved in the process as there is in the conversion of Euler angles.

4. Conclusion

We conclude, therefore, that Euler angles make more sense than Quaternions to pass between networked simulators for the following reasons:

- Networked simulators are currently targeted to "Se-

lective Fidelity" simulators instead of high fidelity simulations. These types of simulators appear to use Euler angle to represent orientation.

- Most simulations compute Euler angles for various purposes. A smaller set of simulations compute only Quaternions.

- Euler angles generate less network traffic than Quaternions

- Higher order dead reckoning models will be solved easier using Quaternions than Euler angles. However, higher order dead reckoning models will usually be used by higher fidelity simulations. These simulations normally have the excess computing capacity to perform the additional required calculations.

- Singularities caused by Euler angles can be overcome within the required accuracy of most simulations.

- Euler angles can be computed for essentially the same computational costs as Quaternions.

Reference

[1] Burchfiel, J. "The Advantages of Using Quaternions Instead of Euler Angles for Representing Orientation," *White Paper ASD-91-001, Third Workshop on Standard foe the Interoperability of Defense Simulations,* Orlando, FL, August, 1990.

[2] "Military Standard (Draft): Protocol Data Units for Entity Information and Entity Interaction in a Distributed Interactive Simulation," Institute for Simulation and Training Publication IST-PD-90-2-Revised, January, 1991.

[3] Roskam, J. "Airplane Flight Dynamics and Automatic Flight Controls, Part I and Part II" Published by Roskam Aviation and Engineering Corporation, 1979.

---

† Some simulators were grouped as, "All kinds of simulators". These types of entries were counted as one simulator type.

This paper investigates the dynamic response of a P-3 aircraft and a light twin-engine turbo-prop to a modeled low-level microburst encounter.

Microburst Model

Background

Fujita[3] first used the term "downburst" to describe a concentrated severe downdraft inducing an outward burst of damaging surface winds. The term "microburst" was used to define a downburst of horizontal extent of no more than 4 km. Early microburst models resolving large-scale effects involved the use of singularities of ideal fluid mechanics, such as sources or circular doublet sheets.[4] In many cases, simple downburst models can serve to accurately represent measured winds for particular windshear events.[5] Such models capture the general three-dimensional downburst wind behavior, yet in some cases may lack the ability to model the series of strong vertical wind gradients identified in recent data.

Caracena[6] suggested in 1982 that the airflow in a microburst may be structured as a vortex ring having downward circulation at the center. Following studies have made use of vortex-ring models to simulate the complex flowfield.[7,8,9] Schultz[10] developed a double-vortex-ring model which accurately simulated the microburst encounter of American Airlines Flight 539 (AA-539) at the Dallas/Ft. Worth Airport (DFW) on 2 August 1985. AA-539 traversed the microburst at an altitude of 2500 ft on a go-around, 110 seconds after Delta Airlines Flight 191 (DAL-191) encountered the microburst on final approach and contacted the ground about one mile short of the runway. The two encounters of the same phenomenon, and the use of DFDRs to enable the determination of winds time histories, have made this windshear event an important test case for microburst modeling. A generalized representation of the encountered microburst is shown in Fig. 1.
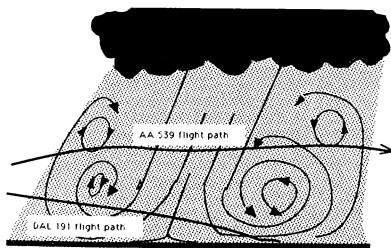
Fig. 1 Generalized microburst structure encountered at Dallas/Ft. Worth on 2 Aug. 1985.

Model Development

A double-vortex-ring model was chosen to simulate the DAL-191 microburst encounter, following the development of such a model by Schultz[10] to represent the winds experienced by AA-539. The analytical representation of the model is documented in Ref. 10 and will not be given here. A general description of the model will be given and the modifications noted.

A diagram of the two vortex rings and their image rings is shown in Fig. 2. The potential flow of the model is composed of vortex-ring filaments embedded in irrotational flow. The viscous core of each is modelled by a continuous vorticity distribution based on a velocity damping factor. The result of the development is a series of algebraic expressions for the velocities induced by the ring filaments. A parameter-estimation scheme was used to minimize the squared-error cost function based on the actual and modeled winds. The parameters of ring radius, ring core radius, position of the ring's center, and the ring's circulation were varied for both rings to achieve the best fit of the winds to those of the flight data.
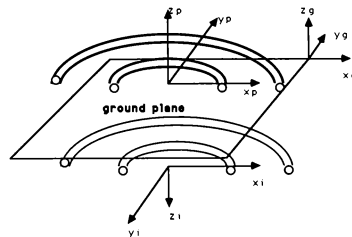

Fig. 2 Double-vortex-ring representation.

The original model was programmed and tested against the horizontal and vertical winds of the AA-539 encounter obtained by the methods of Ref. 2 as documented in Ref. 11. The test case results are shown in Fig. 3. The crosswind component was not considered in this model; this component is not easily modeled by symmetric vortex rings,[10] and has much less impact on the aircraft behavior than the components in the vertical plane. Table 1 shows the final parameters representing the AA-539 windshear.

Current Model

Due to the existence of high surface winds (outflow) in studied microbursts and the relative successes with stagnation-point and point-source models to capture this outflow, a point source and its image were added to the current double-vortex model simulating the microburst encountered by DAL-191. The point source was located at 10,000 ft above and below the ground plane, and the vertical and horizontal strengths were varied in the parameter-estimation scheme
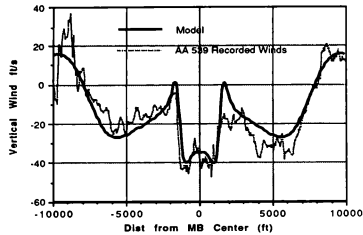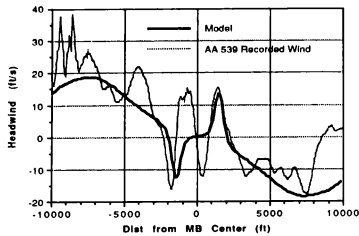
Table 2 Windshear model parameters for DAL-191 encounter near ground level

parameter	large ring	small ring
ring radius, ft	7000	1300
core radius, ft	2004.1	323.9
ring circulation, ft^2/s	431968.8	131571.3
x position, ft	2500.0	300.0
y position, ft	-300.0	1.0
z position, ft	3400.6	800.0
x dir. source strength	1355396 ft^2/s	
z dir. source strength	3049200 ft^2/s	

Fig. 3 Vortex-ring model comparison to recorded flight data (Schultz, Ref. 10).

along with the ring parameters. The results of matching the determined winds of the DAL-191 encounter are shown in Table 2 and in Figs. 4-6. Compared to the two rings in the AA-539 encounter, the small vortex ring is lower to the ground and displaced in the x-direction. Compared to the total-system rms of 16 ft/sec for the AA-539 model, the value for the current model for DAL-191 is 16.5 ft/sec. Vertical and horizontal winds are shown in Fig. 4; the geometry of the modeled microburst is shown in Fig. 5 and the velocity vectors of the flowfield along the desired descent path are shown in Fig. 6.

Fig. 4 DAL-191 recorded winds compared to winds of mathematical model.

Table 1 Windshear model parameters for AA-539 encounter at 2500 feet

parameter	large ring	small ring
ring radius, ft	8503.3	1701.7
core radius, ft	20004.1	323.9
ring circulation, ft^2/s	431968.8	57204.9
x position, ft	0.0	50.0
y position, ft	3350.4	830.9
z position, ft	3400.6	2333.6
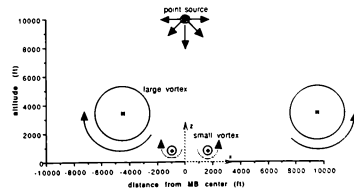
Fig. 5 Microburst model for DAL-191 encounter.
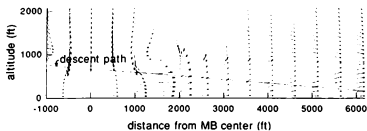
Fig. 6 Velocity vectors along descent path in modeled microburst.

Aircraft Performance Modeling

The concept of aircraft specific energy was used to compare the flight trajectories of various windshear escape strategies. The total energy is defined as the sum of the airmass-relative kinetic energy and the inertial potential energy.

The relationship of the aircraft's motion through a moving airmass relative to earth leads to the development of six coupled differential equations. The development is given in Ref. 12 and will not be included here. The aircraft performance parameters of C_{Lo}, $C_{L\alpha}$, C_{Do}, S, K, W, maximum thrust available, and maximum AoA were input into the equations of motion. C_L and C_D were determined from:

$$C_L = C_{Lo} + C_{L\alpha}\alpha \qquad (1)$$

$$C_D = C_{Do} + K(C_L)^2 \qquad (2)$$

In the equations of motion, the time-derivatives of the vertical and horizontal winds were taken to be zero in the assumption of a steady-state model. The equations are for a point-mass analysis, and longitudinal dynamics are not considered. The equations were solved using a second-order-accurate Euler predictor-corrector scheme with Richardson extrapolation.

Two aircraft were considered for their responses to the DAL-191 microburst windshear. They were the U.S. Navy P-3 (Lockheed L-188) and the U.S. Navy T-44 (Beechcraft King Air H-90). The P-3 is a four-engine turboprop with gross weights in the medium range from 75,000 to 135,000 lbs. The T-44 is a twin-engine turboprop in the light-twin transport category. For a more complete description of the aircraft, see Ref. 13.

Microburst Escape Strategies

Much work has been performed defining the optimal escape maneuvers for airline transport aircraft.[12,14,15,16] The Federal Aviation Administration (FAA) has generated an exhaustive Windshear Training Aid[17] aimed at enhancing the flight safety of modern transport aircraft. It addresses crew training requirements and suggests viable microburst escape strategies. If possible, ground measurements indicating the existence of windshear will alert the transport aircraft in order to avoid an encounter.

The P-3 aircraft has no means of avoidance; it must rely solely on escape. The published guidelines for microburst escape are based upon airline transport aircraft which vary significantly from the P-3. In this study, the results of different escape procedures based on the available flight instruments are compared using the parameters of altitude, airspeed and specific energy. Three flight phases critical to a microburst encounter were considered: the approach to landing, the takeoff, and the on-station phase. Results of the first two are presented here.

Approach to Landing

The landing approach scenario was based on a 3-deg glideslope descent to landing. The microburst center was placed 10,300 feet from the end of the runway. The simulation started 500 horizontal feet before the microburst center on glideslope. This situation closely resembled the condition of DAL-191. The aircraft was exposed to the windshear at t = 0 sec, and the simulation continued for a set time or until ground impact occurred. The results presented here are for a P-3 at 89,500 lbs, a P-3 at 114,000 lbs, and a T-44. More test cases can be found in Ref. 13. Starting aircraft parameters were chosen for the given pitch angle and thrust to maintain a 3-deg glideslope at the target airspeed (V_{ref}). Pitch angle and thrust were maintained until a loss of airspeed equated to V_{ref} minus 20 knots. At that time, the aircraft performed one of the designated escape maneuvers.

Four microburst escape techniques were analyzed for the approach-to-landing encounter. They were:

1) _Constant airspeed_ - Maximum thrust was applied and the pitch angle was set to 0 deg. This pitch angle was maintained until the airspeed equalled V_{ref}. Pitch angle was then adjusted to maintain the airspeed at $V_{ref} \pm 5$ knots.

2) _Constant altitude_ - Maximum thrust was applied and a pitch angle set to obtain a positive rate of climb. Pitch angle was maintained until the target altitude (that at which the maneuver began) was established. Pitch angle was then adjusted to maintain the target altitude $\pm$ 20 ft.

3) _Constant pitch angle_ - Maximum thrust was applied and a pitch angle set and maintained. Pitch angles of 5, 8, 10 and 15 deg were used.

4) _Constant angle of attack_ - Maximum thrust was applied and the pitch angle was adjusted to obtain a given angle of attack. AoA values of 12, 15, and 20 units were used for the P-3 model.

Constraints applied to the escape maneuvers were a pitch-rate limit of 5 deg/sec, a thrust application rate of 50% maximum thrust/sec for the P-3, and a thrust application rate of 20% maximum thrust/sec for the T-44. Maximum pitch angle was limited not to exceed maximum AoA.

Takeoff

The simulation began with the aircraft lifting off 1200 feet prior to the microburst center at the appropriate liftoff speed (V_1). Initial pitch angle was that required to achieve takeoff safety speed (V_2) at 50 feet. Maximum thrust was used for all takeoff simulations. The execution of the particular escape maneuver began when the rate of climb was less than zero, or the airspeed fell below V_2. Presented here are results for a P-3 at 120,000 lbs and a T-44. Four escape methods were considered for the takeoff encounter. They were:

1) <u>Constant airspeed</u> - If the airspeed was less than V_2 at initiation, pitch angle was reduced to 0 deg. If the airspeed was greater than V_2 at initiation, pitch angle was increased. In both cases, pitch angle was manipulated to maintain an airspeed of $V_2 \pm 5$ knots once V_2 was re-established.

2) <u>Constant altitude</u> - Pitch angle was varied to maintain altitude $\pm$ 20 feet about the target altitude at which the escape maneuver began.

3) <u>Constant pitch angle</u> - Pitch angle was held constant throughout the maneuver. Values treated were 5, 10, and 15 deg.

4) <u>Constant angle of attack</u> - Pitch angle was varied to maintain a constant AoA value.

For the above maneuvers, a pitch-rate of 5 deg/sec was used and maximum thrust maintained. Pitch angle was reduced whenever critical AoA was exceeded.

<u>Results</u>

Approach to Landing

As a validation of the model, a three-engine heavy airline transport model was flown through the microburst and compared to the results of the DAL-191 flight. Inputs to the aircraft control were the recorded pitch angle and thrust of the DAL-191 records. The flight-path results were almost identical.

Figures 7-9 depict the response for different escape maneuvers executed by a P-3 at 89,000 lbs gross weight as it flew through the microburst. Plots of altitude, velocity, and specific energy are shown for all cases. The constant-airspeed and 12-units AoA maneuvers were rejected due to resulting flight trajectories below the descent path, as can be seen in Fig. 12. The 20-units AoA and constant-altitude maneuvers were rejected due to resulting speeds below that for flow-separation, as can be noted in Fig. 8. (Maximum lift coefficient was not included in the simulation.) Figure 9 indicates the 5-deg pitch-angle condition resulted in the highest specific energy value at 5000 feet past the microburst center, followed by the 10-deg pitch-angle maneuver.

The next three figures depict the results obtained for different escape maneuvers executed by a P-3 at 114,000 lbs gross weight. As can be noted in Fig. 10, the 12-units AoA, constant-airspeed, and 5-deg pitch-angle maneuvers were
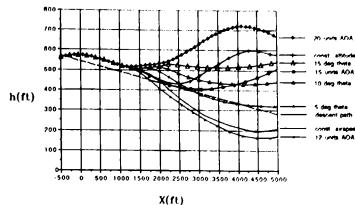

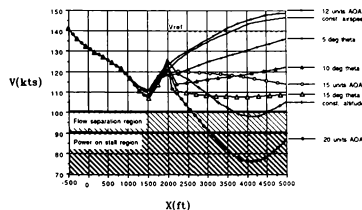Fig. 7 Altitude results for microburst encounter, P-3, approach to landing, 89,000 lbs.

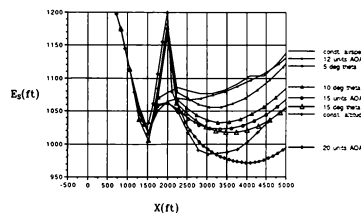Fig. 8 Airspeed results for microburst encounter, P-3, approach to landing, 89,000 lbs.

Fig. 9 Specific energy results for microburst encounter, P-3, approach to landing, 89,000 lbs.

rejected due to trajectories below the desired descent path (the 12-unit AoA strategy resulted in ground impact at 4800 feet). As seen in Fig. 11, the 20-units AoA and constant-altitude cases resulted in airspeeds into the power-on stall region, while the 15-deg pitch angle maneuver gave an airspeed at x = 5000 feet slightly into the flow-separation region. Specific results shown in Fig. 12 indicate that of the remaining strategies, the 15-units AoA and 10-deg pitch angle maneuvers gave the highest values of total energy height.

For both weights of the P-3, of the maneuvers not rejected for low altitude or airspeed,
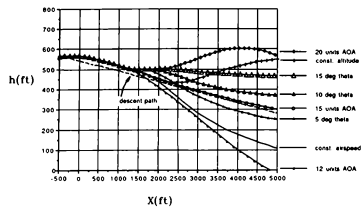
Fig. 10 Altitude results for microburst encounter, P-3, approach to landing, 114,000 lbs.
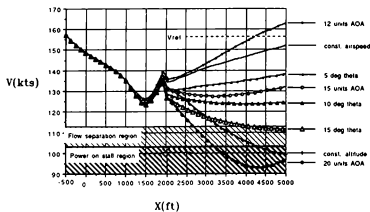

Fig. 11 Airspeed results for microburst encounter, P-3, approach to landing, 114,000 lbs.
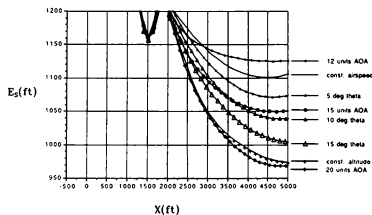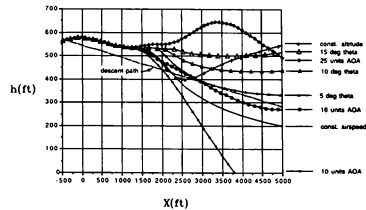

Fig. 12 Specific energy results for microburst encounter, P-3, approach to landing, 114,000 lbs.

the 10-deg pitch-angle strategy achieved a high value of specific energy. Of those not rejected, the 10-deg pitch-angle maneuver resulted in the highest value of altitude.

The approach-to-landing phase was computed for the T-44 aircraft through the modeled microburst windshear. As seen in Fig. 13, the 10-units AoA, constant-airspeed, and 16-units AoA escape maneuvers resulted in trajectories below the descent path. The 25-units AoA and constant-altitude maneuvers were subsequently

rejected due to low airspeeds as can be seen in Fig. 14. Energy plots in Fig. 15 show the remaining maneuvers to be 5, 10, and 15-deg pitch-angle strategies in decreasing order. Large variances in altitude or airspeed between these strategies resolve to be relatively small changes in terms of specific energy.

Fig. 13 Altitude results for microburst encounter, T-44, approach to landing.
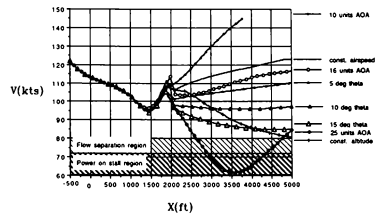

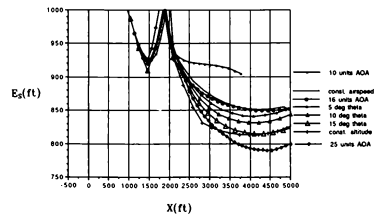Fig. 14 Airspeed results for microburst encounter, T-44, approach to landing.

Fig. 15 Specific energy results for microburst encounter, T-44, approach to landing.

Takeoff Phase

Takeoff escape strategies were tested for the P-3 at 120,000 lbs and for the T-44. Again, shown are plots of altitude, airspeed, and specific energy.

As can be seen in Fig. 16, the strategies of constant airspeed, constant altitude, 5-deg pitch angle, 12-units AoA, and 15-units AoA all result in ground impact from 500 to 2500 feet of the takeoff point. The 10-deg and 15-deg pitch-angle maneuvers survive the encounter; the 15-deg pitch-angle maneuver results in the higher altitude by 4000 feet, but the 10-deg pitch-angle strategy has a flatter descent slope. Figure 17 indicates that neither maneuver results in an airspeed clear of the flow-separation flight regime. The 10-deg pitch-angle case results in an airspeed slightly slower than that for the onset of flow separation, while the 15-deg pitch-angle case is deep within the power-on stall region. The specific energy of the remaining two strategies, as shown in Fig. 18, are almost identical, with the 10-deg case slightly higher.

Included with the P-3 test data at 120,000 lbs is the resulting flight performance when the rotate speed, V_1, is increased 18 knots at the liftoff point 1200 feet before the microburst center for the 10-deg pitch-angle case. The greatest benefit seen from this increase in liftoff speed is an increase in altitude. There is a maximum height gain of 140 feet over that for the standard 10-deg pitch-angle case -- an increase of over 200%. The increase in specific energy is 25% over that for the standard 10-deg pitch-angle case which is significantly greater than the spread of specific energy values between the other escape maneuvers. However, the airspeed of the increased-rotate-speed maneuver is only 5 knots above that for the standard case. This result shows that the additional speed at rotation translates to increased altitude.

The next three figures consider the response of the T-44 aircraft to a takeoff microburst encounter. As indicated in Fig. 19, all escape maneuvers resulted in ground clearance for the 4000 feet beyond the microburst center. The tradeoff of airspeed for altitude is noted in Fig. 20, where it can be seen that the escape-maneuver order is exactly reversed from that of
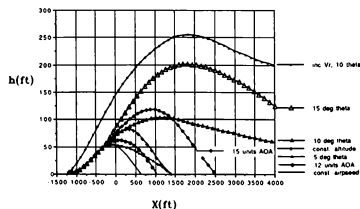

Fig. 16 Altitude results for microburst encounter, P-3, takeoff, 120,000 lbs.
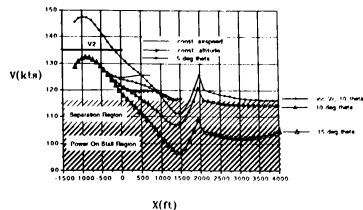

Fig. 17 Airspeed results for microburst encounter, P-3, takeoff, 120,000 lbs.
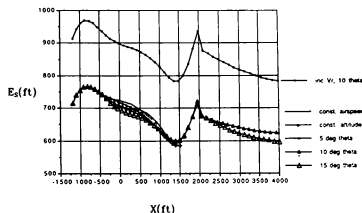

Fig. 18 Specific energy results for microburst encounter, P-3, takeoff, 120,000 lbs.

the previous figure. The 10-deg and 15-deg pitch-angle maneuvers, resulting in moderately increasing values of both altitude and airspeed at x = 4000 feet, appear to be favorable recommendations for optimal escape strategies. Specific energies for the maneuvers are shown in Fig. 21. Values are almost identical for all escape maneuvers.

Other Cases

Various other flight conditions were treated which can be found in Ref. 13 and will only be briefly mentioned here. A P-3 flying at its maximum takeoff gross weight of 135,000 lbs failed to achieve a successful takeoff regardless of the escape maneuver attempted. A P-3 in the approach-to-landing phase at 89,500 lbs operating on three of its four engines resulted in a response similar to that for the heavier P-3. An analysis of the 120,000-lb P-3 during loiter operations resulted in a successful on-station microburst encounter. Additional thrust was not applied until a loss of 40 knots was identified. The P-3 navigated the encounter with an altitude deviation no greater than ± 20 feet and a pitch-angle input no greater than 10 deg.

In a comparison of the effects of aircraft weight, wing loading, and thrust-to-weight ratio, it was found that an increased thrust-to-weight ratio decreased the effect of the wind-
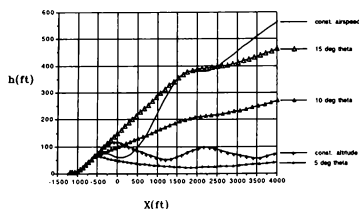
Fig. 19 Altitude results for microburst encounter, T-44, takeoff.
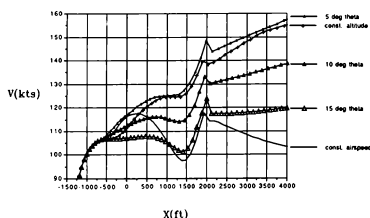

Fig. 20 Airspeed results for microburst encounter, T-44, takeoff.
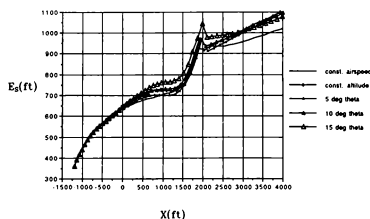

Fig. 21 Specific energy results for microburst encounter, T-44, takeoff.

shear (as expected) and a light wing loading allowed for a more immediate transfer of energy. Weight as an independent parameter was found to be irrelevant.

Stick Force for Off-Trim Airspeed

The expected change in flight control "feel" for the P-3 aircraft is light; flight tests indicate that the stick-force gradient is shallow for this size of aircraft.[18] In the takeoff/approach-flap configuration, at an aft cg of 29% MAC, a stick force of 11 lbs is required for a 30-knot decrease from trim airspeed. Stick force increases to a 16-lb pull with the most forward cg of 16% MAC. Flight test data indicate that stick forces are expected to be almost identical for either the approach-to-landing or takeoff microburst-encounter flight phase.

The P-3 Flight Simulator was used to confirm the results relating to off-speed stick force, as might be experienced during a microburst encounter. The simulator allows one or more flight parameters to be frozen. For the test case, the altitude was frozen at 500 feet and the aircraft was trimmed with maximum power at the reference airspeed. The airspeed was then reset and frozen to the lower airspeed value expected during a microburst windshear. The control force required to maintain a pitch attitude was then measured. With maximum power applied and with a 10-deg pitch attitude, the stick force ranged from a 6-lb pull with an aft cg to a 24-lb pull with a forward cg.

Conclusions

Using the available instruments on board the P-3 and T-44 aircraft, the optimal escape procedure was found to be flying a constant value of pitch angle. Constant-AoA maneuvers sometimes resulted in superior performance, but AoA indication is not recommended due to expected instrument-lag errors during turbulent conditions, and to the fact that AoA indication is seldom used during the approach phase.

Pitch-angle values from 5 to 15 deg were identified as optimal for the approach-to-landing encounter. A pitch-angle value of 10 deg is felt to provide suitable recovery for the P-3 in the configurations tested. A 10-deg pitch angle was also found to work well for the T-44 in the approach-to-landing phase. In the takeoff encounter, a pitch angle of 15 deg was found to work best for the T-44.

Recommendations for the approach-to-landing phase for the P-3 are to set and maintain maximum power while setting a 10-deg pitch attitude. For the takeoff encounter, rotate speed should be increased to 140 knots and a pitch attitude of 10 deg maintained.

The optimum escape procedures are identical for all gross weights. The approach-to-landing escape procedure is identical for three or four operating engines.

The airspeed will decay rapidly and remain abnormally low during a microburst penetration. An elevator force of 5 to 10 lbs can be expected to maintain a 10-deg pitch attitude with full power for the P-3. Flight tests have indicated that the P-3 has no unusual short-period, phugoid, or cross-couple dynamics which would alter these recommendations.

The only different recommendation for the T-44 is the 15-deg pitch angle for takeoff. This difference is probably related to the different flap configuration during takeoff for the T-44, while the P-3 has an identical flap setting for both landing and takeoff flight phases.

In summary, the flight responses of two Navy turboprop aircraft were studied for optimal escape strategies in the event of microburst penetration. The microburst modeled was that encountered by DAL-191 at DFW Airport on 2 August 1985. This study may contribute to the

continued improved modelling of measured micro-bursts for the purpose of aircraft safety and aircrew training.

References

[1]Bach, R.E. Jr., and Wingrove, R.C., "Analysis of Windshear from Airline Flight Data," *Journal of Aircraft*, Vol. 26, Feb. 1989, pp. 103-109.

[2]Wingrove, R.C., and Bach, R.E. Jr., "Severe Winds in the Dallas/Ft. Worth Microburst Meas-ured from Two Aircraft," *Journal of Aircraft*, Vol. 26, Mar. 1989, pp. 221-224.

[3]Fujita, T.T., "Spearhead Echo and Downburst Near the Approach End of a John F. Kennedy Air-port Runway, New York City," SMRP Res. Paper No. 137, Univ. of Chicago, Chicago, IL, 1976.

[4]Zhu, S., and Etkin, B., "Model of the Wind Field in a Downburst," *Journal of Aircraft*, Vol. 22, July 1985, pp. 595-601.

[5]Oseguera, R.M., and Bowles, R.L., "A Simple, Analytic 3-Dimensional Downburst Model Based on Boundary Layer Stagnation Flow," NASA TM-100632, July 1988.

[6]Caracena, F., "Is the Microburst a Large Vortex Ring Imbedded in a Thunderstorm Downdraft?," *EOS Trans. Amer. Geophys. Union*, Vol. 63, 1982, p. 89.

[7]Ivan, M., "A Ring-Vortex Downburst Model for Flight Simulations," *Journal of Aircraft*, Vol. 23, March 1986, pp. 232-236.

[8]Swolinsky, M., "Wind Shear Models for Aircraft Hazard Investigation," AGARD CP-470, May 1989, pp. 2-1 to 2-17.

[9]Grantham, W.J., Roetcisoender, G.G., and Parks, E.K., "DFW Microburst Model Based on AA-539 Data," *Journal of Aircraft*, Vol. 27, Nov. 1990, pp. 917-922.

[10]Schultz, T.A., "A Multiple-Vortex-Ring Model of the DFW Microburst," AIAA Paper 88-0685, January 1988.

[11]"Aircraft Accident Report, Delta Airlines, Inc., Lockheed 1-1011-385-1, N726D, Dallas/Fort Worth International Airport, Texas, August 2, 1985," National Transportation Safety Board, Washington, DC, Report AAR-86/05, Aug. 1986.

[12]Hinton, D.A., "Flight-Management Strategies for Escape From Microburst Encounters," NASA TM-4057, August 1988.

[13]Bobbitt, R.B., "Escape Strategies for Turbo-prop Aircraft in a Microburst Windshear," Mas-ters Thesis, Naval Postgraduate School, Monte-rey, CA, March 1991.

[14]Bray, R.S., "Aircraft Performance and Control in Downburst Wind Shear," SAE Paper 861698, 1986.

[15]Robinson, P.A., and Reid, L.D., "Modeling of Turbulence and Downbursts for Flight Simula-tors," *Journal of Aircraft*, Vol. 27, Aug. 1990, pp. 700-707.

[16]Psiaki, M.L., and Stengel, R.F., "Optimal Aircraft Performance During Microburst Encounter," *Journal of Guidance, Control, and Dynamics*, Vol. 14, March/April 1991, pp. 440-446.

[17]"Windshear Training Aid," Federal Aviation Administration, National Archives, Washington, DC, 1987.

[18]"Lockheed P3V-1 Flying Qualities," Lockheed Aircraft Corp. Report 13133, 19 Nov. 1959.

Observation of chaotic dynamics in vibrating airframes

James Galasyn

Flight Systems Laboratory, Boeing Commercial Airplane Company, Seattle, WA 98124

Two types of commercial jetliner airframes exhibit symptoms of chaotic structural vibrations. Equivocal evidence for the presence of chaotic oscillations is presented: quasi–periodic orbits and double–well potentials similar to those of the buckled beam studied by Moon (1987) are observed. Correlation integrals are computed and fractal dimensions are estimated.

I. INTRODUCTION

Many physical systems can be modeled as a collection of coupled nonlinear oscillators. There is increasing evidence[1] that wide classes of such systems have in common certain dynamics which are labeled *chaotic*. The term "chaotic" should not be taken to mean *random*; instead, it refers to a class of motion with sensitive dependence on initial conditions. Though such motions may appear random*like*, they are, in fact, rigidly deterministic. One goal of the investigator in chaos theory is to find the underlying determinism and model it as a set of coupled ordinary differential equations. Often these equations are of remarkably low order; in many instances[2] high–dimensional behavior can be reduced to a one–dimensional mapping. Such maps are easily computed and can produce high–fidelity, extremely fast models for simulation.

Irregular vibrations are observed in many physical phenomena, including small–amplitude vibrations of air-borne structures. In turbulent airflow structural vibrations can become uncomfortable. An active control system can help, but only if the structural state is understood. Control–law development would be greatly simplified if a high–fidelity, time–domain simulation of structural vibration modes were available. This paper describes preliminary efforts toward this end.

In Sec. II the theory of chaotic dynamics is briefly described and previous results are reviewed. Sec. III describes the experimental arrangement. Results are discussed in Sec. IV, and conclusions are summarized in Sec. V.

II. THEORY

Recent developments in the field of chaotic dynamics have produced new methodologies for analyzing irregular data that confound linear system theory. Of great usefulness in approaching such data is the phase–space portrait $\{x, \dot{x}\}$, which plots the velocity *vs.* position trajectory of the data vector $x(t)$. This plot can show where stable and unstable equilibria are to be found. Often only one variable is available for measurement, however, and the other must be computed by integration or differentiation. Alternately, a powerful method can be applied to the single time–series measurement: phase–space reconstruction, or the "embedding space" method.[3] With this technique it is possible to track the motion through a pseudo phase–space $\{x(t), x(t-T), x(t-2T),...x(t-nT)\}$, for $n \in \{1, 2, 3,...\}$ and T an arbitrarily chosen lag. Often useful is the two–dimensional embedding space $\{x(t), x(t-T)\}$, which approximates the phase–space $\{x, \dot{x}\}$ with surprising accuracy. Much insight into the nature of the system under consideration can be gleaned from the reconstructed phase–space portrait.

Another method is then applied to the phase–space reconstruction in order to determine the nature of the motion. If the trajectory $\{x, \dot{x}\}$ is digitally sampled by $x_n \equiv x(t_n)$ and $y_n \equiv \dot{x}(t_n)$, then its locus in the phase plane is a two–dimensional mapping

$$x_{n+1} = g(x_n, y_n) \quad (1)$$
$$y_{n+1} = h(x_n, y_n).$$

If there is a forcing function with period f, and the sampling times t_n are chosen to be $t_n = n/f$, then (1) is a Poincare map. This is a useful diagnostic tool for identifying various classes of motion in $x(t)$. Specifically, the nature of the system's *attractors* can often be determined with the Poincare map. These attractors may be of the classical types (fixed points, limit cycles or toroids), or the attractors may be *strange*. The Poincare map is most helpful in distinguishing among the various motions, but has the drawback that for chaotic motions a large number of high–resolution samples must be taken to reveal fine structure. This limitation can be severe, and for this reason analog recording devices are often favored over digital instruments.

These methods have been used successfully by a number of researchers in reducing high–dimensional behavior to low–dimensional mappings. Two problems in particular are closely related to that of the airframe vibrating under turbulent inputs: aeroelastic or panel flutter, and the buckling elastic beam.

Dowell[4] has done extensive work in characterizing the flow over a buckled elastic plate. Moon[5] has conducted detailed studies of the periodically–forced, buckled beam. Both have reported chaotic behavior in the observed structural oscillations (Fig. 1).

Throughout this paper the airframe is modeled as a beam. Nonlinearities are assumed to arise from several sources, most significantly from highly energetic airflow over elastic surfaces and from nonlinear structural fixity and effectiveness. Bending of the beam is assumed to be of small enough amplitude that geometric nonlinearities can be neglected. Holmes[6] models the situation as a system of three coupled first–order differential equations:

$$\dot{x} = y$$
$$\dot{y} = -\gamma x + \frac{1}{2}(1 - x^2) - A_0\omega^2\cos z \qquad (2)$$
$$\dot{z} = \omega,$$

where x is displacement (nondimensional modal amplitude), γ is the damping coefficient, ω is the forcing frequency, and z is phase, chosen to be $z = 0$ for convenience. This simple nonlinear system displays chaotic phase–space trajectories (Fig. 2). Experimental bending–strain *vs.* strain–rate portraits match the computed trajectories well. Even more remarkably, the experimental phase–space vector $\{x(t), x(t+T)\}$, reconstructed from

the measurement of a single strain variable only, bears striking qualitative similarity to the computed trajectories. Moon and Li[7] also find good correspondence between the *fractal dimension*, D_F, of the computed Poincare map and that of the measured, reconstructed vibration data, with $D_F = 2.5$.

Chaotic trajectories never close or repeat, but they *are* bounded within well–defined regions; these regions have fractal boundaries.[8] The chaotic motion tends to fill up a region of space, and the fractal dimension measures how much space a strange attractor fills. If the dimension of the attractor is noninteger, this can be taken as an indication that the attractor is strange.[9]
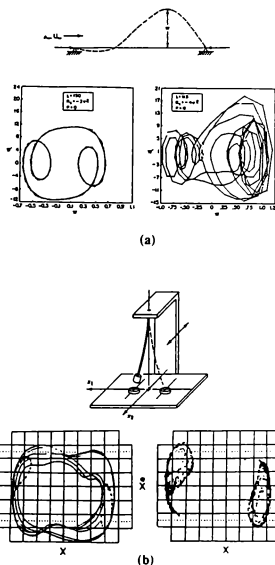

(a)

(b)

Fig. 1. Two vibrating systems that exhibit chaotic behavior. (a) Aeroelastic flutter of a buckled elastic plate. *Left:* Periodic oscillations. *Right:* Chaotic oscillations (from Dowell, 1982). (b) Buckled elastic beam. *Left:* Quasi–periodic vibrations of two frequencies, with ω_1/ω_2 rational. Poincare section shows evenly spaced dots on an ellipse. *Right:* Poincare section showing fractal structure in the chaotic regime (from Moon, 1987).

Estimating the fractal dimension D_F is computationally intensive for systems of higher than second order,[10] and many attempts have been made to find more efficient algorithms for estimating D_F. The method employed here is due to Grassberger and Procaccia.[11] A correlation integral $C(g)$ is defined:

$$C(g) \equiv \lim_{N \to \infty} \frac{1}{N^2} \sum_{i,j} H(g, D_{ij}) \qquad (3)$$

where D_{ij} is the distance between points x_i and x_j and H is the Heaviside step function

$$H(g, D) = \begin{cases} 0, & g < D; \\ 1, & g > D. \end{cases} \qquad (4)$$

The distance measure D_{ij} need not be Euclidean, and for computational speed it can be chosen to be a simple function, *e.g.*, $|x_i - x_j|$.[12]

$C(g)$ increases from zero for small g (*i.e.*, small length scales) to one for large g. Furthermore, Grassberger and Procaccia demonstrate that for small g,

$$C(g) \propto g^{\upsilon}. \qquad (5)$$

The correlation integral thus has a power–law dependence on the exponent υ. Moreover, υ is so closely related to D_F that it can be *taken for D_F*; it is one of a number of different dimension measures that fall into the category of "fractal dimension." When υ is noninteger, *i.e.*, fractal, the attractor is strange, and the motion is chaotic.

If only a single variable has been observed, the time series $x(t)$ is then embedded in increasingly higher dimensions, $D_E \in \{2, 3, 4, \ldots\}$. If a low–dimensional attractor is present the slopes D_C of $\log C(g)$ *vs.* $\log g$ may converge over some range of g. The value of this slope is taken to be a fractal dimension of the attractor.

This fractal dimension can exhibit strong dependence on one or more system parameters. It has been shown by Grebogi, *et al.*.[13] that as a system parameter is varied, the fractal basin boundaries can experience rapid changes: jumping to new positions or changing from smooth manifolds to fractal manifolds and back. Grebogi, *et al.*, refer to such sudden changes as *metamorphoses*.

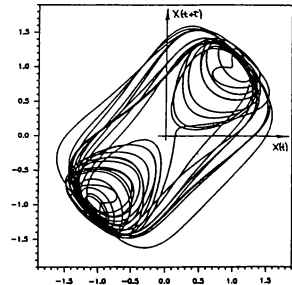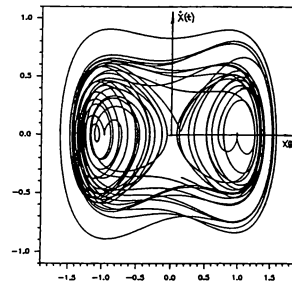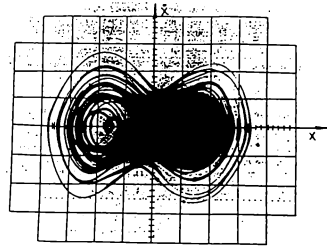### III. EXPERIMENTAL ARRANGEMENT

Fig. 2. Chaotic trajectories in the buckled elastic beam. *Top:* Experimental observation of velocity *vs.* position. *Middle:* Orbits $\{x, \dot{x}\}$ computed from Equation (2). *Bottom:* Pseudo–phase–space portrait of $\{x, \dot{x}\}$ reconstructed from computed $x(t)$ (from Moon, 1987).

Off–the–shelf data sets from previous flight tests were examined for evidence of chaotic behavior. Lateral accelerations N_Y (in body coordinates) from three points along two different types of airborne commercial jetliners (designated VK001 and NA001) were recorded as digitally sampled time series data. Three flight conditions were studied: airframe VK001 flying in turbulent air, VK001 flying through smooth air with rudder frequency sweeps, and NA001 flying through smooth air with rudder "dwells" (single–frequency tone bursts). The sampling rate was 100 samples per second (sps) for turbulent air conditions and 200 sps for smooth air conditions with sweeps and dwells.

Commanded rudder sweeps to VK001 were linear ramps from 0 Hz to 8 Hz over a period of 90 seconds (Fig. 3a). Altitude was 15,000 feet, calibrated airspeed was 300 knots.

Commanded rudder dwells to NA001 were single–frequency tone bursts at 1.00 Hz, 2.00 Hz, 2.25 Hz, 2.50 Hz, 2.75 Hz, 3.00 Hz, 3.25 Hz, 3.50 Hz, 4.00 Hz and 4.50 Hz. Each burst lasted approximately 15 seconds. Altitude was 13,000 feet, calibrated airspeed was 280 knots.

Each data set $N_Y(t)$ was then lagged against itself, i.e., the vector $\{N_Y(t), N_Y(t-T)\}$ was constructed. Takens[14] suggests that T can be chosen arbitrarily, and values of $2 < T < 60$ samples generally produced the most revealing portraits for the airframe vibration data.

The data sets $N_Y(t)$ were then embedded in increasingly higher dimensions $D_E \in \{2, 3, 4,...9\}$. Correlation integrals $C(g)$ and slopes $D_C = \Delta \log C(g) / \Delta \log g$ were computed for each phase–space embedding.

Finally, correlation dimensions D_C were estimated for regions of the input frequency.

Certain obstacles intervened in the acquisition of vibration measurements, chief of which was the dearth of flight–test data for single–frequency sinusoidal rudder inputs (rudder dwells). Dwell tests are generally only run on the ground, as a check for structural hysteresis. The few available airborne dwell tests were of limited duration, generally about 15 seconds or 3000 samples; it is difficult to distinguish between the Poincare map of quasiperiodic behavior and that of chaotic behavior with so few samples. Much of the fine fractal structure, if any exists, will remain latent and invisible.

Further loss of fine structure occurs when the Poincare section sampling rate is not "stroboscopically"

related to the driving frequency f by an integer fraction.[15] Hence, with a fixed sampling rate of 200 sps, only the integer factors of 200 (e.g., 1 Hz, 2 Hz, 4 Hz, etc.) for driving frequency are likely to produce recognizably fractal maps.

The sampling rate also affects the computation of correlation integrals. There is no general method for determining the best choice of sampling rate, and if it is too high, spurious correlations can be introduced.[16] Again, there is no choice but to use the available 200 sps rate.

Another limitation is imposed on measurements of accelerations N_Y: The 10–bit digitized signal does not provide enough resolution to observe fine–structure variations; this amounts to digital quantization error that badly obscures any fractal detail which may be present in the Poincare map. This effect can be ameliorated somewhat by Spline interpolation, which yields reasonably smooth phase–space portraits. Interpolation is dangerous for estimating fractal dimension, however, and can produce spurious, almost–integer values of D_F.[17] Interpolated data sets are therefore not used in this paper for dimension calculations.

Atmospheric noise is also present in the rudder dwell and sweep tests, and it is difficult to determine if broadband noise in the spectra is due to turbulent airflow shocks or to chaotic frequency contributions. It has been argued[18] that for motion on a strange attractor, noise will not eradicate the fractal structure, but will smear it on length scales smaller than that of the noise. The Grassberger–Procaccia method of estimating the correlation exponent υ may show a region of convergence for short length scales. This reflects the space–filling motion of small–amplitude noise, which is not deterministic. Schaffer[19] prescribes that spuriously high correlations will be observed in the slopes of log $C(g)$ vs. log g for small g. This region may obscure lower dimensional scaling regions of deterministic motion, making them narrower.

The above drawbacks notwithstanding, the data are more than adequate for computing correlation integrals and hence fractal dimensions of motion in the yaw (N_Y) plane. The minimum number of state–space variables is determined after the method of Takens. All computations were performed on an IBM 386 workstation with the Dynamical Systems, Inc. DSI and DSII software packages.[20]

Data sets were available for two different airframe types, and both were examined separately. The VK001 rudder sweeps are useful for qualitatively determining which regions of input frequency display unusual behavior. The absence of dwell tests, however, limits the quantitative analysis that can be performed in the regions of interest: Calculation of correlation integrals and spectra are both sensitive to changes in forcing frequency; further-
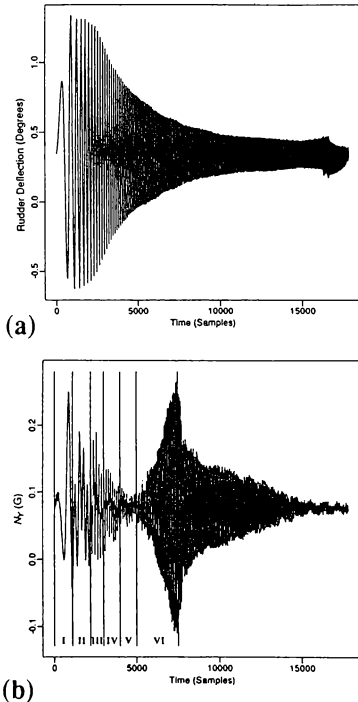

(a)

(b)

Fig. 3. Linear rudder–frequency sweeps from 0 to 8 Hz. (a) Rudder response with low–pass filter characteristic. (b) Airframe response at aft body bulkhead. Trace has been divided into six regions according to qualitative behavior.

more, the number of samples recorded in each region is too small to generate a meaningful Poincare map. Still, useful observations can be made.

Time–series traces for rudder deflection and N_Y are shown in Fig. 3. The responses, as expected, are highly nonlinear. For reasons which will be discussed below, the time series data for rudder sweeps have been divided into six regions, based on certain qualitative behaviors observed there.

Consider Region I in Fig. 3b, shown in greater detail in Fig. 4a. The response to low–frequency rudder excitation is reasonably linear; the response tracks the input with little amplitude or phase change. Transients can be seen riding the forcing frequency and decaying away.

Contrast this behavior with that of Region II. The airframe response shows a noticeable wobble, which in Region III takes on the appearance of several frequencies beating against one another. Transients have died away. In Region IV higher frequency features become apparent, and in Region V the signal has become irregular in appearance. In Region VI, the response becomes periodic or quasi–periodic as it approaches the primary body–bending mode for VK001, $f_0 = 3.3$ Hz.

At this point it is revealing to examine the pseudo–phase–space reconstructions of these regions (Fig. 4b). The signal $N_Y(t)$ is plotted against $N_Y(t-T)$ for $T \in \{4, 12, 48\}$. These values for T are arbitrarily chosen to produce the most comprehensible portraits.

The Region I trajectory reflects the quasi–linear behavior of the low–frequency airframe response. This is most clearly demonstrated by the $T = 48$ portrait. The motion is clockwise from the center. Here, the elliptical spiral structure results from a sinusoid of increasing amplitude plotted against its derivative. Transients are spaced around the circumference and appear as tight "knots" that relax with time. If the system were linear, we would expect to see the trajectory settle into a steady–state periodic orbit (a limit cycle) after the amplitude had become constant and transients had died away.

But in Region II the situation changes rapidly. Instead of stable elliptical orbits, several interlocking, "figure–eight" orbits appear. These strongly suggest the presence of a two–well potential function, where none was visible before. It is possible that a *metamorphosis* has occurred: a stable, fixed–point equilibrium has split

into one unstable and two stable equilibria similar to the situation shown in Fig. 2.

As the forcing frequency f increases, we see in Region III that the "potential wells" expand and become more defined, but in Region IV they begin to merge, or collapse, until in Region V complex deformations occur,
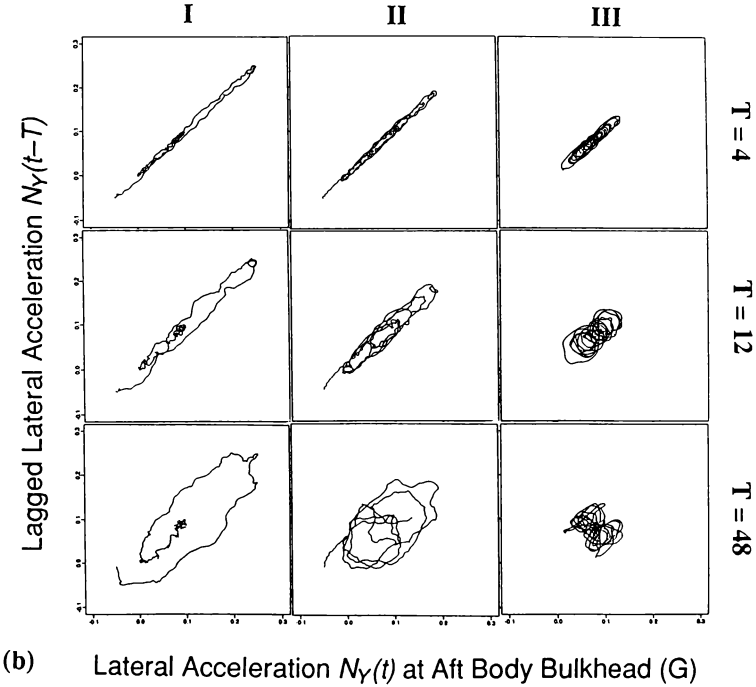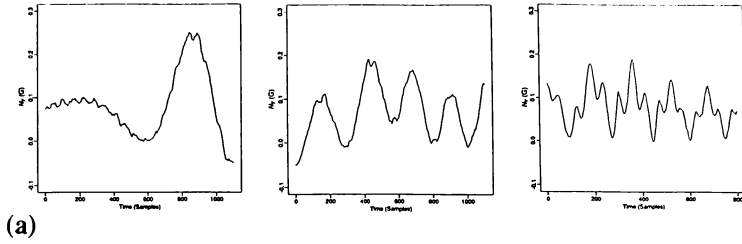

(a)

(b)

Lateral Acceleration $N_Y(t)$ at Aft Body Bulkhead (G)

Fig. 4. Rudder–sweep data for Regions I–III of Fig. 3. (a) Time–series traces. (b) Phase–space reconstructions $\{N_Y(t), N_Y(t-T)\}$, $T \in \{4, 12, 48\}$.

accompanied by an overall attenuation of amplitude.

Region VI illustrates the behavior near the primary body–bending mode $f_0 = 3.3$ Hz. The structure resonates strongly in this region, and limit–cycle–like orbits move outward, clockwise from the center. They also exhibit flat pass–band responses that appear rather like quanta or
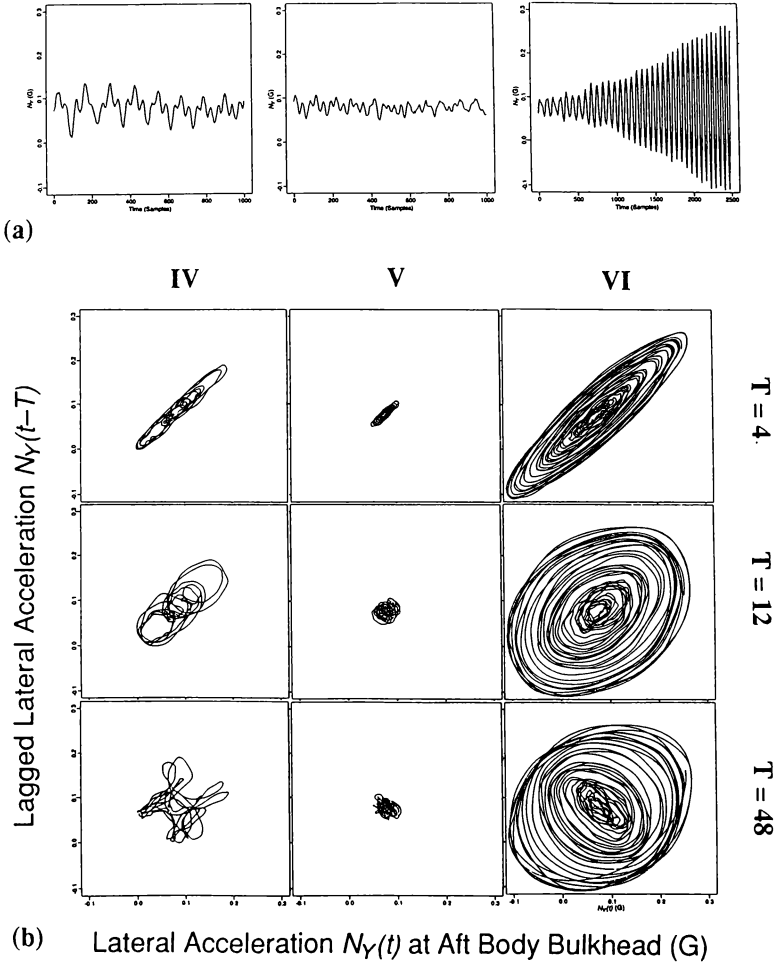

(a)

(b) Lateral Acceleration $N_Y(t)$ at Aft Body Bulkhead (G)

Fig. 4 (cont'd). Rudder–sweep data for Regions IV–VI of Fig. 3. (a) Time–series traces. (b) Phase–space reconstructions $\{N_Y(t), N_Y(t-T)\}$, $T \in \{4, 12, 48\}$.
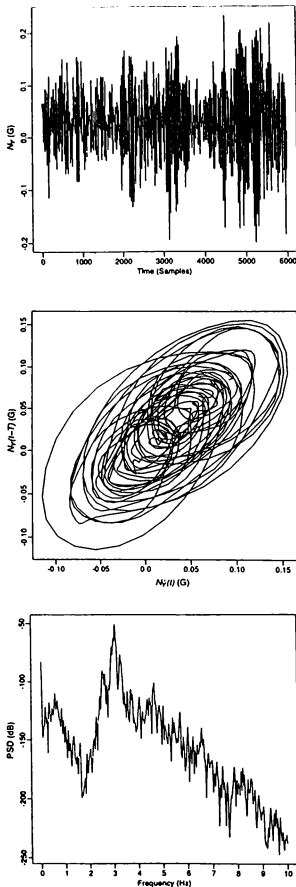
Fig. 5. Response of VK001–type airframe to turbulent airflow. *Top:* Time-series trace. *Middle:* Pseudo–phase-space portrait, with reconstruction lag $T = 4$ samples. *Bottom:* Power spectral density.

energy levels.

It is important to understand that the variation in these tangled trajectories may not be the result of random fluctuations. If the system is truly chaotic, in fact, these orbits are not stochastic at all, but deterministic, sharp kinks and doublings–back included. If the two–well structure is apparent under random inputs, then we may have more faith in the robustness of the double–well potential model. It is therefore instructive to examine the phase–space reconstruction when the inputs to the system *are* stochastic.

Consider Fig. 5. What is most striking about the vector $\{N_Y(t), N_Y(t-T)\}$ in turbulent air conditions is its qualitative similarity to the double–well potentials of Dowell and Moon (Figs. 1,2). The two–well structure is robust even with a random driving function, and this lends credence to the idea that we are not simply observing a noisy period–two motion.

As noted above, chaotic orbits never close or repeat, but instead tend to fill up a region of the phase space. The boundaries of this region depend on critical values of system parameters. The above exercise tracked the qualitative characteristics of the attractor structure as a function of driving frequency. But it did not tell us if the attractor is indeed *strange*, and therefore, if the motion is truly chaotic. A good quantitative measure for strangeness is to estimate the correlation dimension D_C. If the slope, D_C, of $\log C(g)$ vs. $\log g$ is noninteger, then the dimension of the trajectory is fractal and the motion is chaotic.

The caveat here is that the input frequency is not constant, and therefore the trajectory will fill a different region of the phase–space than it would for a dwell test; this may produce spurious estimates of D_C. Keeping this in mind, application of the Grassberger–Procaccia method to rudder–sweep data yields noninteger values for D_C. As we would expect, at low driving frequencies (Regions I and II), the dimension of the airframe vibration is almost exactly integer, $D_C \cong 2.0$. With increasing f, the dimension increases to values intermediate between 2.0 and 3.0. The average of the four noninteger values is approximately 2.45, which is in agreement with Moon's numerical estimate of $D_C \cong 2.5$ for (2).[21]

This value implies that at this fluid velocity and compressive load condition, the airframe vibration be-

293

havior, from steady–state to chaotic, can be captured with a three–variable state–space. It seems likely that with the proper rudder–dwell tests, the coefficients of (2) can be determined. The airframe structural response in the yaw plane would then be completely described by a single 3 x 3 matrix.

D_C was also estimated for the turbulent air condition in Fig. 5, with $D_C \cong 3.6$. This value must also be treated with skepticism, as noise will tend to increase the dimension estimate for length scales smaller than the noise (see discussion of Fig. 9 below). The phase–space portrait (Fig. 5) does not have the appearance of a random walk, however, and if we make the assumption that some determinism is at work here, the $\Delta D_C \cong 1.0$ implies the

existence of another state variable when speed and altitude are considered. Airflow quantities such as dynamic pressure, angle of attack, or strain measurements of fin side–loads might be appropriate choices for this fourth state variable. Only dynamic pressure has been examined as of this writing, and the bandwidth of the pressure–port data was too low to be useful for high–speed phase–space analysis. Alpha vanes or strain gauges may prove to be of high–enough fidelity for this purpose. As above, a compact representation may be possible in a low–dimensional state–space, with the airframe response in the N_Y plane described by a 4 x 4 matrix.

The rudder sweep tests were not ideal for most quantitative analyses. Numerical examinations were
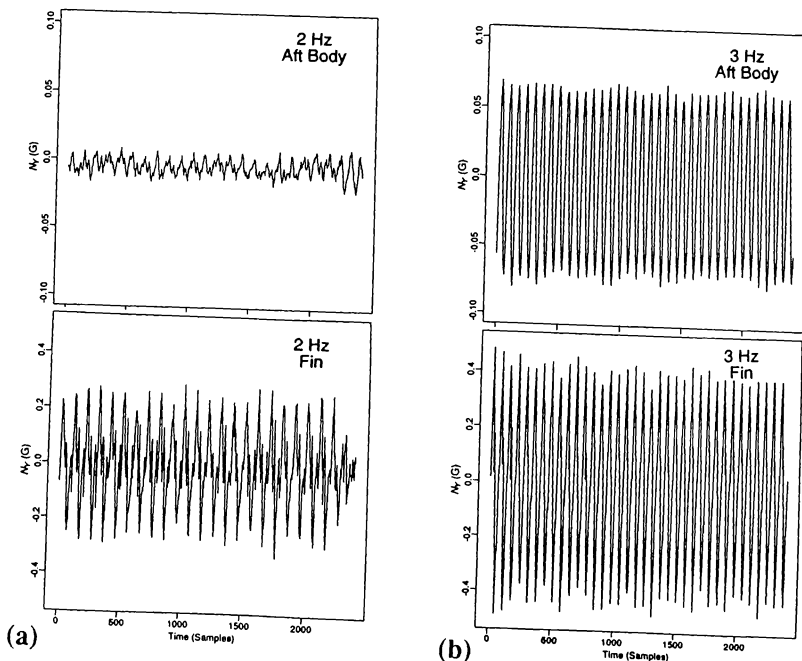

Fig. 6. NA001 response to sinusoidal rudder inputs. (a) Response to 2–Hz tone burst. *Above:* Lateral acceleration at aft body bulkhead. *Below:* Lateral acceleration at fin. (b) Response to 3–Hz tone burst. *Above:* Lateral acceleration at aft body bulkhead. *Below:* Lateral acceleration at fin.

more readily performed on the measurements from the NA001 rudder–dwell tests. For brevity, only the response to two values of driving frequency will be discussed.

Fig. 6 shows time–series traces of the NA001 airframe response to 2–Hz and 3–Hz excitations. Accelerations are shown for the aft body bulkhead and the fin. Phase–space reconstructions of these data are shown in Fig. 7, and power spectra are shown in Fig. 8.

The phase–space portraits for NA001 body responses are generally less dramatic than those of VK001 (Fig. 7). Figure–eight orbits are not observed; rather, the motion is suggestive of periodic or quasiperiodic behav-

ior. The fin response, however, bears some similarity to the VK001 body responses, and to the Fig. 1 trajectories. Both double–well potentials (Fig. 7a) and quasiperiodic orbits (Fig. 7b; compare with Fig. 1b) may be present.

Fig. 8 shows spectra for body and fin vibrations. Above about 8 Hz all responses are dominated by noise, most likely from turbulent atmospheric shocks. The noise floor for the body response is roughly 125 dB down from that of the same region in the fin data.

Most prominent in all spectra are two spikes: one at the rudder driving frequency and one at about 0.25 Hz. This corresponds to the Dutch–roll mode, a contribution
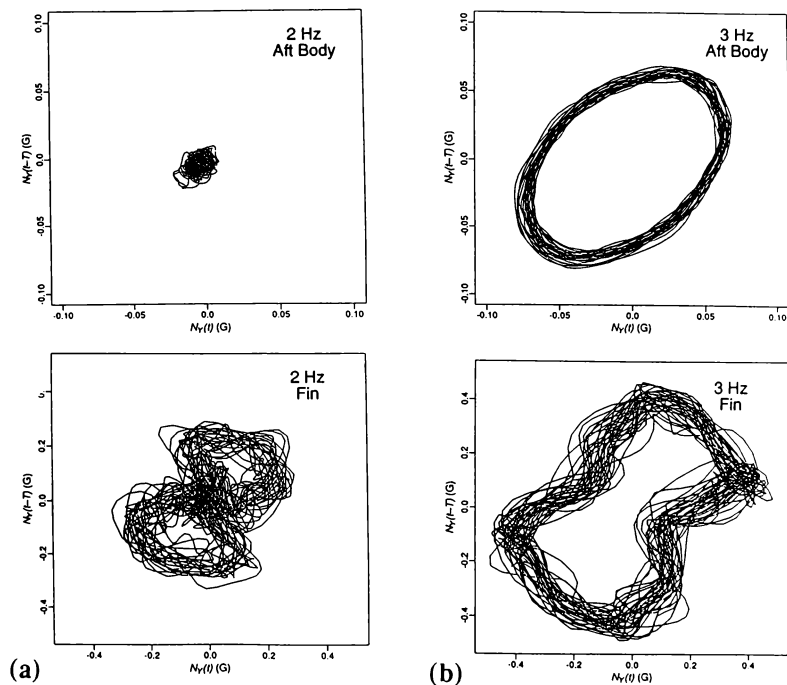

Fig. 7. NA001 response to sinusoidal rudder inputs. (a) Response to 2–Hz tone burst. Pseudo–phase–space portraits of lateral acceleration at aft body bulkhead (*above*) and at fin (*below*). (b) Response to 3–Hz tone burst. Pseudo–phase–space portraits of lateral acceleration at aft body bulkhead (*above*) and at fin (*below*).

cies, the spectra give little insight into the nature of the motion. The available data suggest noisy quasiperiodic motion; such motion often precedes the onset of chaotic behavior as a system parameter (*e.g.*, driving frequency) is varied. If the driving frequencies had been anharmonically related to the Dutch–roll mode, more obviously chaotic motions might have been observed.

Fig. 9 provides more insight. Here, the slope of log $C(g)$ are computed for embedding dimensions $D_E \in \{2, 3,...9\}$ for the body and $D_E \in \{2, 3,...10\}$ for the fin. When the rudder is driven at 2 Hz, the slopes converge to $D_C \cong 5.0$ for $D_E > 5$ and $-6 < \log g < -5$. For longer length scales no further correlations are observed. For the 3–Hz input the slopes converge at a much larger scale $-4 < \log g < -3$ to a value of $D_C \cong 1.3$. This region of scale invariance suggests the existence of an attractor, and the noninteger value suggests the attractor is strange.

Similar regions of convergence appear in the fin vibrations, but here, the higher–dimensional convergence zone occurs for longer length scales. The lower–dimensional zone converges to $D_C \cong 2.6$ for the 2–Hz input and $D_C \cong 1.3$ for the 3–Hz input. This decrease in dimension would be expected if the motion were changing from chaotic to quasiperiodic, and therefore from more to less space–filling, as the forcing frequency is varied. The phase–space portraits in Fig. 7 suggest such a change in behavior.

The higher–dimensional zone of convergence, with $D_C \cong 5.0$ for the smallest length scales, is probably the result of noise. As noted above, observational noise has the effect of producing two regions of convergence: one at small scales, which reflects the space–filling property of the noise, and one at longer scales, which represents the dimension of the deterministic motion. If the signal–to–noise ratio is not too low, the two zones will be distinct, and the true dimension can be discerned.

That the noisy regions are separated from the lower–dimensional regions may give us confidence in the estimated values of D_C. But once again, the sample size becomes an issue: the smaller the number of observations, the higher the estimated value of D_C.[22] With fewer than 3000 samples for the dwell tests, the ostensible scaling region may seriously overestimate the correlation dimension, and this effect is exacerbated by the presence of noise, which further narrows the region and also may increase the apparent dimension. Until long dwell test are

performed, the present estimates of D_C must be treated with skepticism.

V. SUMMARY AND CONCLUSIONS

Airframe responses to rudder frequency sweeps and dwells were examined using phase–space reconstruction. For airframe VK001 this method suggested the presence of double–well potentials similar to those encountered in the buckled elastic beam problem. The motion around these potentials appeared chaotic. Fractal dimensions were estimated using the correlation integral method of Grassberger and Procaccia. These were found to be in agreement with the dimension of the buckled beam vibrations reported by Moon. Similar results obtained for the dimension of the NA001 fin, but were not observed for the NA001 body.

The motivation for these investigations was to examine the possibility of producing a high–fidelity, high–speed, time–domain structural vibration model for digital simulation. Such a model would assist in the design of a yaw damper control law for actively suppressing irregular airframe oscillations in turbulent airflows.

The embedding–space method provides a powerful new tool for analyzing irregular motions. The Poincare map holds great potential for reducing high–dimensional behavior to low–dimensional mappings; a one– or two–dimensional recursion relationship can capture the important features of many–dimensional motion with extraordinary precision. Such mappings require little computation and can provide real–time simulation on the smallest of workstations.

Preliminary results indicate that chaotic processes are present in airframe vibrations. Data were of insufficient digital resolution and sample size to generate meaningful Poincare sections. More detailed analysis was not possible given the amount of off–the–shelf data presently available from previous flight tests.

These difficulties underscore the need for flight tests specifically designed to characterize the behavior of suspected chaotic regions more accurately. Very long dwell tests at many closely–spaced rudder frequencies are necessary to collect enough samples to build a solid Poincare map. The input frequencies should not be harmonically related to the Dutch–roll mode. The sampling rate should be chosen so as not to introduce artifactual

correlations. The SNR should be high. These measurements should be made with high–fidelity accelerometers capable of more than 10 bits of resolution,; analog devices could be employed to reveal fine fractal structure. Wind–tunnel testing could prove to be equally valuable.

ACKNOWLEDGEMENTS

[1] See, for example, A. Brandstäter, J. Swift, H. Swinney and A. Wolf, Phys. Rev. Lett. **51**, 1442 (1983); R. Harrison and D. Biswas, Nature. **321**, 394 (1986).

[2] For example, L. Glass, M. Guevara, J. Belair and A. Shrier, Phys. Rev. Lett. **29**(3), 1348 (1984).

[3] N. Packard and J. Crutchfield, Phys. Rev. Lett. **45**(9), 712 (1980).

[4] E. Dowell, J. Sound Vib. **85**(3), 333 (1982); E. Dowell, J. Appl. Mech. **53**(1), 5 (1986).

[5] F. Moon, *Chaotic Vibrations for Scientists and Engineers* (John Wiley & Sons, New York, 1987).

[6] P. Holmes, Philos. Trans. R. Soc. London A **292**, 419 (1979).

[7] F.C. Moon and G. –X. Li, Physica **17D**, 99 (1985).

[8] F.C. Moon, Phys. Rev. Lett. **53**(10), 962 (1984).

[9] J. Farmer, E. Ott and J. Yorke, Physica **7D**, 153 (1983).

[10] H. Greenside, A. Wolf, J. Swift and T. Pignataro, Phys. Rev. A **25**, 3453 (1982).

[11] P. Grassberger and I. Procaccia, Phys. Rev. Lett. **50**(5), 346 (1983).

[12] F. Moon, *Chaotic Vibrations*, 216.

[13] C. Grebogi, E. Ott and J. Yorke, Phys. Rev. Lett. **56**(10), 1011 (1986).

[14] F. Takens in D. Rand and L. Young (eds), *Dynamical Systems and Turbulence* (Springer–Verlag, Berlin, 1981).

[15] F. Moon, *Chaotic Vibrations*, .

[16] W. Schaffer, *Dynamical Software DSI*, 1.73 (Dynamical Systems, Inc., Tucson, Arizona, 1988).

[17] W. Schaffer, *Dynamical Software DSII*, 6.21 (Dynamical Systems, Inc., Tucson, Arizona, 1988).

[18] P. Grassberger and I. Procaccia, Phys. Rev. A **29**(2), 975 (1984).

[19] W. Schaffer, *ibid.*, 1.75.

[20] W. Schaffer, *ibid.*

[21] F. Moon, *Chaotic Vibrations*, 232.

[22] W. Schaffer, *ibid.*, 1.66–1.73.

[23] J. R. "BoB" DoBBs, Ride Quality Internal Memoranda, Boeing Commercial Airplane Company, Renton, WA, 1990.

ASSESSMENT OF MICROBURST MODELS FOR DOWNDRAFT ESTIMATION

Dan D. Vicroy *
NASA Langley Research Center
Hampton, Virginia

Abstract

Wind shear is considered by many in the aviation industry to be one of their major safety issues. The National Aeronautics and Space Administration has been conducting research in the development of forward-looking, airborne wind shear detection system technologies. Doppler RADAR and LIDAR are two of the technologies being tested to provide this capability. Both measure the Doppler shift of reflected light or radio waves from the aerosols, rain drops and other debris in the air, to determine the line-of-sight relative velocity of the air. An inherent limitation of this type of system is its inability to measure velocities perpendicular to the line-of-sight. This limitation can result in a significant underestimate of the magnitude of the wind shear hazard. One solution to this "line-of-sight" limitation of Doppler type sensors is to use a theoretical or empirical model of a microburst to estimate the perpendicular velocities from the measured line-of-sight values. This paper is a summary of an analytical study to assess the effectiveness of three microburst models in estimating the downdraft from horizontal velocity measurements. The paper discusses the development of the models and their characteristics. The models are tested at different stages in the life cycle of a microburst.

Symbols

F wind shear hazard index

g gravitational acceleration

r radial coordinate

t time

u horizontal wind component, tailwind positive

$\dot{u}$ rate of change of horizontal wind component

V true airspeed

w vertical wind component, updraft positive

z vertical coordinate

Introduction

Wind shear is considered by many in the aviation industry to be one of their major safety issues. Numerous accidents and incidents have occurred that were attributed to low-altitude wind shear, which can be found in a variety of weather conditions such as gust fronts, sea-breeze fronts, and mountain waves. Hazardous wind shear is most often associated with the convective outflow of thunderstorms known as microbursts. The microburst is a strong localized downdraft, which causes a significant outflow as it impacts the ground. The hazard of a microburst encounter arises from the rapid shift from head-wind to tail-wind as the airplane penetrates the microburst outflow, which in turn reduces the airplane's airspeed. This is accompanied by the downdraft component of the microburst that reduces the airplane's rate of climb. The general effect on the airplane is a rapid loss of energy from which it may not have enough altitude, airspeed or thrust to overcome.

The National Aeronautics and Space Administration, in a joint effort with the Federal Aviation Administration, has been conducting research in the development of forward-looking, airborne wind shear detection system technologies. Forward-look systems give advanced warning of the presence of wind shear and thus provide the flight crew with the time needed to avoid the affected area or escape from the encounter. A fundamental requirement for such a wind shear detection system is the ability to estimate reliably the magnitude of the wind shear hazard that would be experienced by an airplane if it were to continue along the line-of-sight. Doppler RADAR and LIDAR are two of the technologies being tested to provide this capability. Both measure the Doppler shift of reflected light or radio waves from the aerosols, rain drops and other debris in the air, to determine the line-of-sight relative velocity of the air. An inherent limitation of this type of system is its inability to measure velocities perpendicular to the line-of-sight. The presence of a microburst can be detected by measuring the divergence of the horizontal velocity profile, yet, the inability to measure the downdraft can result in a significant underestimate of the magnitude and spatial extent of the hazard.

One solution to this "line-of-sight" limitation of Doppler type sensors is to use a theoretical or em-

pirical model of a microburst to estimate the perpendicular velocities from the measured line-of-sight values. This paper is a summary of an analytical study to assess the effectiveness of three microburst models in estimating the downdraft from horizontal velocity measurements. The paper discusses the development of the models and their characteristics. The models are compared at different points in the life-cycle of a microburst using a high fidelity atmospheric simulation.

The paper will begin with a brief description of a wind shear hazard index known as the "F-factor" to establish the accuracy with which the downdraft needs to be determined. The microburst simulation data base against which the downdraft models will be evaluated will then be described. This will be followed by a description of the three microburst downdraft models and the method of analysis. Analysis results will then be presented, followed by concluding remarks.

Wind Shear Hazard Index

The magnitude of the hazard posed by a microburst to an airplane can be expressed in terms of the "F-factor".[1] The F-factor is a hazard index that represents the rate of specific energy loss due to wind shear. For straight and level flight the F-factor can be expressed as:

$$F = \frac{\dot{u}}{g} - \frac{w}{V} \qquad (1)$$

Positive values of F indicate a performance-decreasing situation, and conversely, negative values indicate a performance-increasing condition. The F-factor is directly related to the climb gradient or rate-of-climb capability of the airplane.[1] For example, a F value of 0.2 would indicate a loss in climb gradient capability of 0.2 radians (11.5 deg). If an airplane had a maximum climb angle capability of 10 degrees, it would be unable to maintain level flight in that wind shear environment. An average F value of 0.1 or greater, extending a range of 1 kilometer or more, is considered hazardous to landing or departing airplanes.[1]

As mentioned in the introduction, the Doppler type wind shear sensors can only measure the line-of-sight divergence of the wind and therefore can only determine the first term of the F-factor equation ($\dot{u}/g$). The inability to determine the second term of the equation can result in a significant underestimate of the magnitude of the microburst hazard. Figure 1 shows the magnitude of the F-factor due to the downdraft at various airspeeds. At an airspeed of 130 knots, a downdraft greater than 6.7 m/s (13 knots) would exceed the 0.1 F-factor hazard threshold. For landing or departing airplanes at low altitude, the
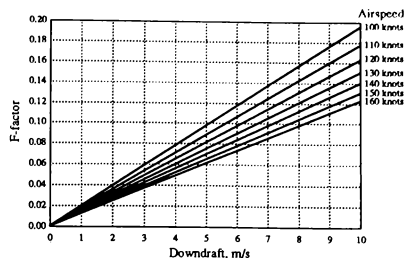
Fig. 1. Sensitivity of the hazard index to the downdraft magnitude at various airspeeds.

vertical component of F can exceed half the total F-factor.

Microburst Simulation Model

The data set upon which the various downdraft models were evaluated was generated with the Terminal Area Simulation System (TASS) high-fidelity atmospheric simulation model. TASS is a time-dependent, multi-dimensional, nonhydrostatic, numerical cloud model that has been used extensively in the study of microbursts.[2,3] It is initiated with the observed atmospheric conditions existing prior to microburst development and outputs a three-dimensional time history of radar reflectivity, winds, temperature, pressure, water vapor, rainwater, snow, hail, and cloud water. TASS has been validated against both ground-based and airborne measurements of different microburst events.[3,4] Three-dimensional data sets can be generated at horizontal grid resolutions of 200 meters. Axisymmetric data sets can be generated at grid resolutions as fine as 20 meters. Although TASS is much too complex to be useful as a real-time downdraft estimation model, it is very useful for generating the data sets necessary to evaluate such models.

The microburst simulation used in this paper was axisymmetric with a 20-meter grid resolution. The data set extended from the microburst core to 4000 meters radially and from the ground to 600 meters vertically. TASS was initiated with the atmospheric conditions measured before the August 2, 1985, Dallas-Fort Worth microburst event. Four different times in the microburst simulation were selected, each two minutes apart. Figure 2 shows the wind vector plots for the four time periods selected. The first was just before the downdraft impacted the ground, which was at 9 minutes into the microburst simulation. The second was just after the downdraft hit the ground and began
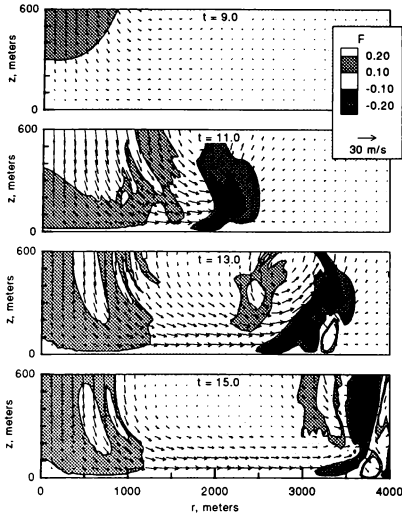
Fig. 2. Wind vectors and F-factor contours of the microburst simulation at four times (t = 9, 11, 13 and 15 minutes).

Fig. 3. F-factor contours computed without vertical winds.

to spread out, which was approximately the time of maximum horizontal shear. The third was at a point when the outflow vortex ring was well defined, and the last, near the end of the life cycle.

Contour plots of F-factor for an airplane flying level at 130 knots are also shown in figure 2. Figure 3 shows the same data sets with the F-factor contours computed without the vertical winds. The magnitude and spatial extent of the detectable hazard is clearly diminished. This further illustrates the need for some means of determining the magnitude of the downdraft.

Microburst Downdraft Models

Three downdraft models were developed for estimating the vertical winds of a microburst from radial wind measurements. The three models represent different levels of sophistication. A description of the three models follows.

Linear Model

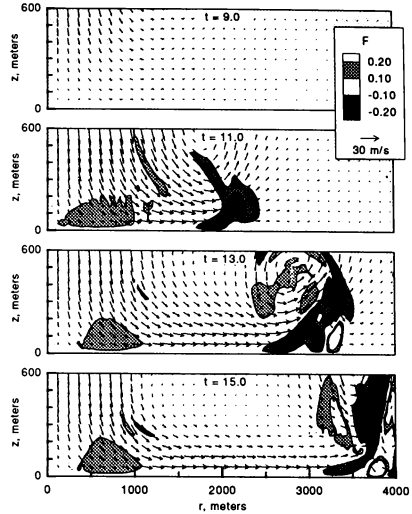The "linear model" is the simplest of the three models tested. It is based primarily on the principle

of conservation of mass, which can be expressed in cylindrical coordinates as:

$$\frac{\partial u}{\partial r} + \frac{\partial w}{\partial z} + \frac{u}{r} = 0 \qquad (2)$$

Since the first two terms of the equation can be obtained from the radial velocity measurement, the vertical gradient of the vertical wind can be determined. If the vertical wind is assumed to be zero at the ground and vary linearly with altitude (i.e. $\partial w/\partial z$ =constant), then the vertical wind can be computed from:

$$w = \frac{\partial w}{\partial z}z = -z\left(\frac{\partial u}{\partial r} + \frac{u}{r}\right) \qquad (3)$$

The validity of the linear approximation can be seen in figure 4, which shows how the downdraft at the center of the TASS microburst varies with altitude for each of the four times. The assumption of linearity appears to be reasonable near the center, particularly at altitudes below 400 meters. At the higher altitudes the linearity assumption begins to break down. Figure 5 shows the vertical variation of the downdraft at a radius of 2000 meters. The linearity assumption is not nearly as valid at this location. This is primarily
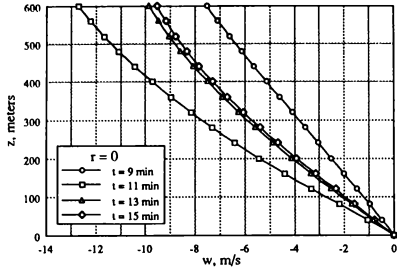
Fig. 4. Vertical wind variation with altitude at the center of the microburst for t = 9, 11, 13 and 15 minutes.
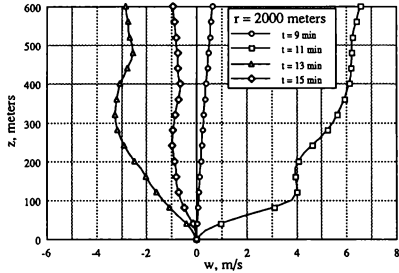

Fig. 5. Vertical wind variation with altitude at a radius of 2000 meters for t = 9, 11, 13 and 15 minutes.

due to the outflow vortex ring. For the 11 minute case, the outflow vortex core was at a radius of about 1800 meters, resulting in updrafts at the 2000 meter radius. For the 13 minute case, the vortex core was beyond the 2000 meter radius, resulting in an increased local downdraft. The degree to which these nonlinearities introduce errors into the downdraft calculation will be determined in the analysis.

Empirical Model

As the name implies, the parameters for this model are based on measurements of several microburst events. The empirical model is a modified version of a microburst model developed by Oseguera and Bowles[5] for research in wind shear escape procedures. The modifications made to the Oseguera/Bowles' model are discussed in detail in reference 6. The empirical model is an axisymmetric, steady-state model that uses shaping functions to satisfy the mass continuity

equation and simulate boundary layer effects.

The mass continuity equation (eq. 2) can be satisfied by solutions of the form:

$$u = f(r)p(z) \tag{4}$$

$$w = g(r^2)q(z) \tag{5}$$

provided,

$$\frac{\partial [rf(r)]}{\partial r^2} = \frac{\lambda}{2}g(r^2) \tag{6}$$

$$\frac{\partial q(z)}{\partial z} = -\lambda p(z) \tag{7}$$

where:

$f(r)$ radial shaping function of the horizontal wind velocity, m/sec;

$g(r^2)$ radial shaping function of the vertical wind velocity;

$p(z)$ vertical shaping function of the horizontal wind velocity;

$q(z)$ vertical shaping function of the vertical wind velocity, m/sec;

λ scaling factor, 1/sec.

The characteristic shape of the radial shaping functions is shown in figure 6. Figures 7 and 8 show the radial profiles of the horizontal and vertical winds from the TASS simulation data at an altitude of 200 meters. The shaping functions are used to approximate the characteristic profile of the microburst winds. The shaping functions appear to compare well with the TASS profiles at the first two event times. But, the radial profiles at the latter time show significant variation due to the vortex ring. As with the linear model, these vortex ring aberrations may introduce significant errors into the downdraft estimate.

The vertical shaping functions are shown in figure 9. The shaping functions compare well with the TASS data shown in figure 4, but not in the area of the vortex ring, as shown in figure 5.

The equations for the shaping functions are:

$$f(r) = \frac{\lambda r}{2}e^{\left[\frac{2-(r^2/r_m^2)^\alpha}{2\alpha}\right]} \tag{8}$$

$$g(r^2) = \left[1 - \frac{1}{2}\left(\frac{r^2}{r_m^2}\right)^\alpha\right]e^{\left[\frac{2-(r^2/r_m^2)^\alpha}{2\alpha}\right]} \tag{9}$$
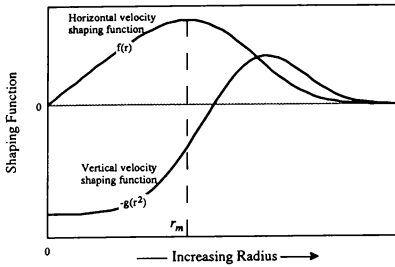
303

Fig. 6. Characteristic variation of radial shaping functions.
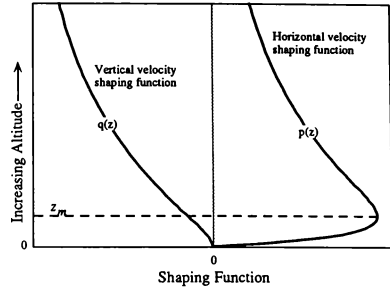
Fig. 9. Characteristic variation of vertical shaping functions.
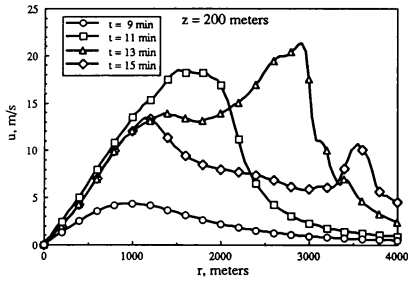
Fig. 7. Radial variation of the horizontal wind at an altitude of 200m for $t = 9$, 11, 13, and 15 minutes.
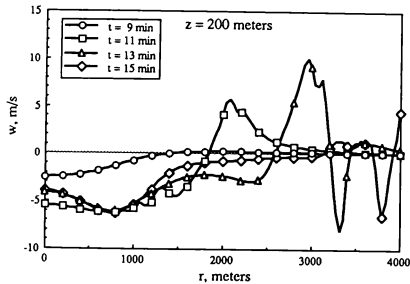
Fig. 8. Radial variation of the vertical wind at an altitude of 200m for $t = 9$, 11, 13, and 15 minutes.

$$p(z) = e^{c_1(z/z_m)} - e^{c_2(z/z_m)} \tag{10}$$

$$q(z) = -\lambda \left\{ \frac{z_m}{c_1} \left[e^{c_1(z/z_m)} - 1 \right] \right. $$
$$\left. - \frac{z_m}{c_2} \left[e^{c_2(z/z_m)} - 1 \right] \right\} \tag{11}$$

where

$$c_1 = -0.15 \tag{12}$$

$$c_2 = -3.2175 \tag{13}$$

The values for c_1 and c_2 were obtained by curve fitting data from several microburst events. These shaping functions yield the following equations for the horizontal and vertical winds.

$$u = \frac{\lambda r}{2} \left[e^{c_1(z/z_m)} - e^{c_2(z/z_m)} \right] e^{\left[\frac{2-(r^2/r_m^2)^\alpha}{2\alpha} \right]} \tag{14}$$

$$w = -\lambda \left\{ \frac{z_m}{c_1} \left[e^{c_1(z/z_m)} - 1 \right] - \frac{z_m}{c_2} \left[e^{c_2(z/z_m)} - 1 \right] \right\}$$
$$\times \left[1 - \frac{1}{2} \left(\frac{r^2}{r_m^2} \right)^\alpha \right] e^{\left[\frac{2-(r^2/r_m^2)^\alpha}{2\alpha} \right]} \tag{15}$$

The empirical model is fully defined through four model variables: the radius and altitude of the maximum horizontal wind (r_m and z_m respectively), a shaping function variable (α), and a scale factor (λ).

304
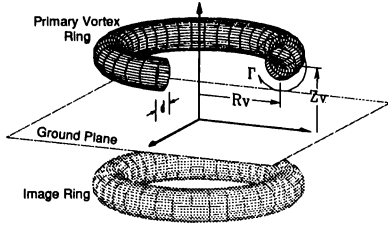
Fig. 10. Ring-vortex model.

Ring-Vortex Model

The ring-vortex model is a theoretically derived model based on the assumption that the flow field generated by a vortex ring near a flat plate is similar to that of a microburst. Ring-vortex models have been used in the past as a somewhat simple method to simulate microbursts.[7,8,9] Models using multiple ring-vorticities have shown good correlation with airborne measured microburst winds.[8]

The ring-vortex model has a primary vortex ring located above the ground and a mirror image ring located equidistant below the ground. The mirror image ring is used to satisfy the no-flow through the ground boundary condition. Figure 10 shows the ring-vortex model and the variables that define it. The ring-vortex model is defined by four model variables: the radius and altitude of the primary vortex ring (R_v and Z_v, respectively), the diameter of the viscous core (d), and the circulation strength (Γ).

The derivation of the velocity equations for the ring-vortex model is discussed in detail in references 7 and 8 and will be omited here for brevity. The ring-vortex velocity equations are:

$$u = \frac{-\Gamma}{2\pi r}\left[\frac{ac\,(d_1 - d_2)^4\,(z - Z_v)}{\sqrt{d_1{}^3 d_2{}^3}\,(d_1 + d_2)^2\left(b + \frac{2c\sqrt{d_1 d_2}}{d_1 + d_2}\right)^2}\right.$$

$$+ \frac{3a\,(d_1 - d_2)^2\,(z - Z_v)}{d_1 d_2\,(d_1 + d_2)\left(b + \frac{2c\sqrt{d_1 d_2}}{d_1 + d_2}\right)}$$

$$- \frac{ac\,(d_3 - d_4)^4\,(z + Z_v)}{\sqrt{d_3{}^3 d_4{}^3}\,(d_3 + d_4)^2\left(b + \frac{2c\sqrt{d_3 d_4}}{d_3 + d_4}\right)^2}$$

$$\left. - \frac{3a\,(d_3 - d_4)^2\,(z + Z_v)}{d_3 d_4\,(d_3 + d_4)\left(b + \frac{2c\sqrt{d_3 d_4}}{d_3 + d_4}\right)}\right] \tag{16}$$

$$w = \frac{-\Gamma}{2\pi r}\left\{\frac{ac\,(d_1 - d_2)^3\left[d_1{}^2\,(r + R_v) - d_2{}^2\,(r - R_v)\right]}{\sqrt{d_1{}^3 d_2{}^3}\,(d_1 + d_2)^3\left(b + \frac{2c\sqrt{d_1 d_2}}{d_1 + d_2}\right)^2}\right.$$

$$+ \frac{a\,(d_1 - d_2)\left[3d_1{}^2\,(r + R_v) + 2d_1 d_2 R_v - 3d_2{}^2\,(r - R_v)\right]}{d_1 d_2\,(d_1 + d_2)^2\left(b + \frac{2c\sqrt{d_1 d_2}}{d_1 + d_2}\right)}$$

$$- \frac{ac\,(d_3 - d_4)^3\left[d_3{}^2\,(r + R_v) - d_4{}^2\,(r - R_v)\right]}{\sqrt{d_3{}^3 d_4{}^3}\,(d_3 + d_4)^3\left(b + \frac{2c\sqrt{d_3 d_4}}{d_3 + d_4}\right)^2}$$

$$\left. - \frac{a\,(d_3 - d_4)\left[3d_3{}^2\,(r + R_v) + 2d_3 d_4 R_v - 3d_4{}^2\,(r - R_v)\right]}{d_3 d_4\,(d_3 + d_4)^2\left(b + \frac{2c\sqrt{d_3 d_4}}{d_3 + d_4}\right)}\right\}$$

$$\tag{17}$$

where:

$$d_1 = \sqrt{(r - R_v)^2 + (z - Z_v)^2} \tag{18}$$

$$d_2 = \sqrt{(r + R_v)^2 + (z - Z_v)^2} \tag{19}$$

$$d_3 = \sqrt{(r - R_v)^2 + (z + Z_v)^2} \tag{20}$$

$$d_4 = \sqrt{(r + R_v)^2 + (z + Z_v)^2} \tag{21}$$

and

$$a = 0.788 \tag{22}$$

$$b = 0.25 \tag{23}$$

$$c = 0.75 \tag{24}$$

The values for a, b, and c were obtained from reference 7.

The ring-vortex model has singularities along the vertical axis and the vortex core. The singularity along the vertical axis is removed by setting the velocities equal to the limit solution.

$$\text{for}\quad r = 0 \begin{cases} u = 0 \\ w = \dfrac{\Gamma R_v^2}{2}\left(1/d_2{}^3 - 1/d_4{}^3\right) \end{cases} \tag{25}$$

The singularity at the vortex core is eliminated by multiplying the velocities by a vicious core damping

305

factor (ζ). The viscous core damping factor is the same as in reference 8.

$$\zeta = 1 - e^{\left\{ \dfrac{-(d_{min}/R_v)^2}{\left[k_1\left(\frac{d}{R_v}\right)+k_2\left(\frac{d}{R_v}\right)^2+k_3\left(\frac{d}{R_v}\right)^3\right]^2} \right\}} \qquad (26)$$

where d_{min} is the minimum of d_1 and d_2; and:

$$k_1 = 0.4215 \qquad (27)$$

$$k_2 = 0.0822 \qquad (28)$$

$$k_3 = -0.0969 \qquad (29)$$

Method of Analysis

The objective of the downdraft models is to compute the radial distribution of the vertical wind, as shown in figure 8, from the radial distribution of the horizontal wind, as shown in figure 7. The ability of each of these models to meet this objective was determined by computing the downdraft from the horizontal velocity profile at each altitude, from 0 to 600 meters, in 20 meter steps. The mean and standard deviation of the downdraft estimation error was computed at each altitude and over the total altitude range. The error was defined as:

$$W_{error} = w - w_{model}$$

Each model computed the downdraft in a slightly different manner. The linear model first computed the radial gradient of the horizontal velocity profile ($\partial u/\partial r$) at each data point using a 2 point central difference scheme. The vertical velocity component was then computed from equation 3.

The empirical model and the ring-vortex model both used a multi-variable search to find the values of the model variables that yielded the best fit to the horizontal velocity profile at each altitude. Using those values, the downdraft profile was then computed for that altitude. The empirical model employed a three variable search for r_m, α and λ (z_m was set equal to 60 meters for this study). The ring-vortex model used a four variable search for R_v, Z_v, d and Γ.

Analysis Results

The analysis of the results consisted of summarizing the downdraft estimate error statistics for each
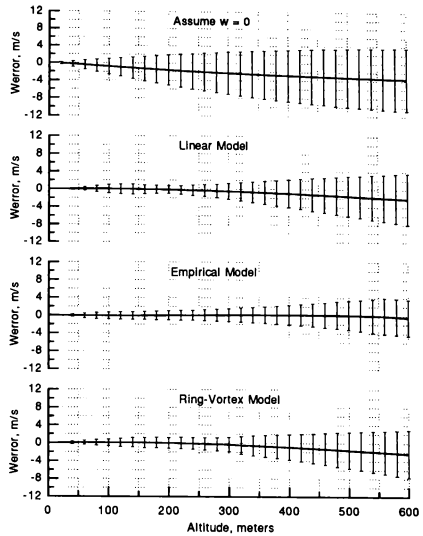
Fig. 11. Mean and standard deviation of the downdraft estimate errors with altitude ($t = 11$ minute case).

model, characterizing the errors and proposing methods to improve the model performance.

Figure 11 shows the mean and standard deviation of the downdraft estimate errors from the three models for the 11 minute case. The errors are shown for each altitude at which a downdraft profile was estimated. Also shown in the figure is the error that results from assuming no downdraft ($w = 0$). The errors increased with altitude for all of the models. This was anticipated for the linear model. As previously mentioned, the linearity assumption breaks down at the higher altitudes. The 9, 13 and 15 minute cases showed similar trends, but varied in magnitude.

Figure 12 shows the total mean downdraft error and standard deviation, from 0 to 600 meters altitude, at each of the four times. The empirical model showed the best overall results for all but the 9 minute case, where the linear and ring-vortex model did very well.

The radial downdraft profiles generated by the models were reviewed at several altitudes to study the basic characteristics of the models. Figure 13 shows the fit of the empirical and ring-vortex models to the horizontal velocity profile, at 300 meters altitude, for

306
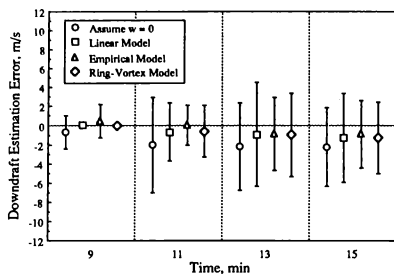
Fig. 12. Total mean and standard deviation of the downdraft errors across all altitudes for t = 9, 11, 13 and 15 minutes.
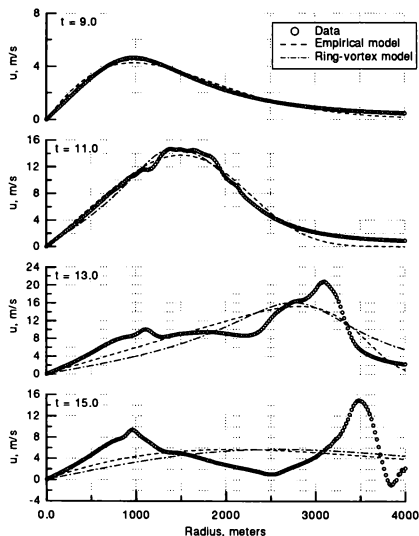
Fig. 13. Fit of the empirical and ring-vortex models to the horizontal velocity profile (z = 300 meters).

Fig. 14. Comparison of the downdraft profiles generated by the three models (z = 300 meters).

the linear model. As discussed earlier, this is due to the basic assumptions inherent in the model. Smoothing the horizontal velocity profile before computing the downdraft may reduce this problem. The second observation is that the empirical and ring-vortex model couldn't replicate the multiple peaks in the horizontal velocity profile, which were generated by the outflow vortex ring. This is particularly evident in the 13 and 15 minute cases. Since the models couldn't accurately replicate the horizontal velocity profile, the resultant downdraft estimate suffered.

The outflow vortex ring does not contribute to the downdraft over an area large enough to be considered a performance hazard to the airplane. The area in which the downdraft is most significant is near the core of the microburst. There are a variety of ways in which this outflow vortex problem could be alleviated. One example would be to use a weighting factor to influence to fit of the model more toward the core of the microburst. Another method would be to limit the fit to a percentage of the radius of the first peak in the horizontal velocity profile. The first peak in the horizontal profile is generally associated with the downdraft of the microburst. The second and third peaks can be associated with the outflow vortex. This

each of the four times. Figure 14 shows the resultant downdraft estimate, including the linear model estimate. Two observations can be made from these two figures. The first observation is that slight variations in the slope of the horizontal velocity profile resulted in significant variations in the downdraft generated from

Fig. 15. Fit of the empirical and ring-vortex models to the horizontal velocity profile with fit limited to twice the radius of the first peak ($z = 300$ meters).
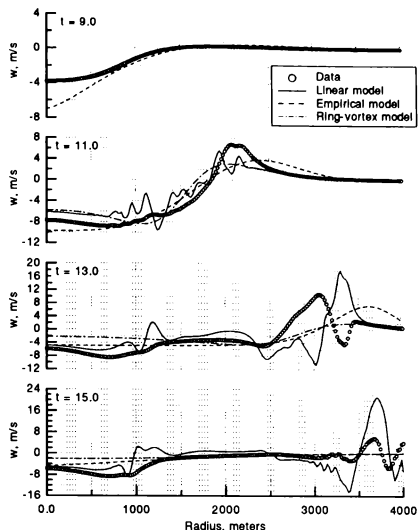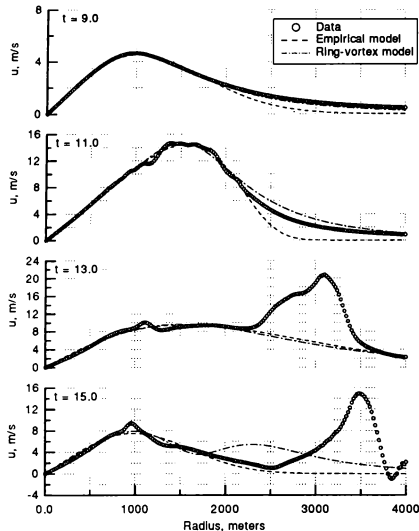
Fig. 16. Comparison to the downdraft profiles generated by limiting the horizontal fit to twice the radius of the first peak ($z = 300$ meters).

is particularly true for this axisymmetric microburst simulation. Asymmetric microbursts may not be as clearly delineated.

Figure 15 shows the results of limiting the fit of the model to twice the radius of the first peak in the horizontal velocity profile. The new fit was with the same data set as in figure 13. Near the core of the microburst the fit is significantly improved. The resultant downdraft computation is shown in figure 16. The downdraft estimate near the core is improved over that shown in figure 14 for both the empirical and the ring-vortex model.

Figure 17 shows the total mean downdraft error and standard deviation for the limited fit procedure, over the full altitude range (0 to 600 meters), at each of the four times. Since the downdraft near the core of the microburst is of primary interest, the error statistics were computed over a limited range near the core (0 to 2000 meters radius), as apposed to the full data range (0 to 4000 meters) as in figure 12. This reduced the large standard deviations produced by the outflow vortex and resulted in an improved summary of the errors in the area of interest. For

comparison, the summary results presented in figure 12 were recomputed for the same range as in figure 17 (0 to 2000 meters radius) and are shown in figure 18. The ring-vortex model improved the most with the limited fit procedure. The effect on the empirical model was small.

The ring-vortex and empirical models gave the best overall results. The ring-vortex model provided the best results for the 9 and 15 minute cases. The empirical model produced the best results for the 11 and 13 minute cases. The 11 minute case is near the time of maximum shear and is perhaps the most critical from a hazard perspective. The linear model worked well for all the cases, at altitudes below 200 meters, particularly near the core of the microburst.

Concluding Remarks

The primary objective of this study was to assess the effectiveness of three microburst models in estimating the downdraft from horizontal velocity measurements. The results indicate that any of the three models would substantially improve the hazard estimate generated with the downdraft neglected.
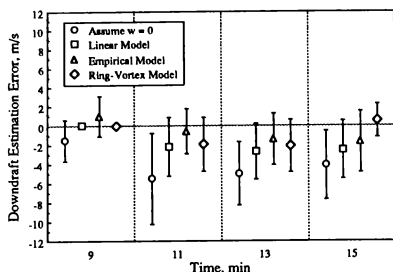
Fig. 17. Total mean and standard deviation of the downdraft error over the range of 0 to 2000 meters radius, with the limiting fit procedure.
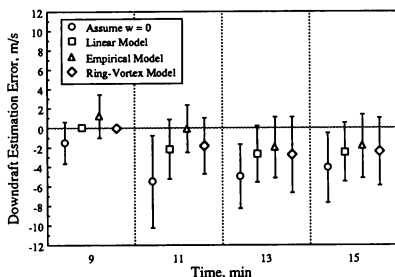
Fig. 18. Total mean and standard deviation of the downdraft error over the range of 0 to 2000 meters radius (horizontal fit over full radius).

The linear model works well for altitudes below 200 meters and near the microburst core. Above 200 meters the errors tend to be amplified. The linear model was the simplest of the three models tested. Smoothing of the horizontal velocity profile data before computing the downdraft may improve the linear model performance at the higher altitudes.

The ring-vortex and empirical models gave the best overall results. The ring-vortex model was the most complex, requiring four variables to define the model. The empirical model used three model variables. Both the ring-vortex and the empirical model were adversely affected by the variations in the horizontal velocity profile from the outflow vortex ring. This problem can be partially elevated by limiting the model fit or weighting the fit to the region near the core of the microburst where the most significant por-

tion of the downdraft occurs.

There is considerable research remaining to assess fully the potential of these microburst models to estimate the downdraft. This study was limited to a symmetric microburst simulation. The ability of these models to determine the downdraft of asymmetrical microbursts remains to be seen. There are also on-board computation and system implementation issues that have yet to be addressed.

References

1. Bowles, Roland L.: *Reducing Windshear Risk Through Airborne Systems Technology.* 17th Congress of the International Congress of Aeronautical Sciences, Stockholm, Sweden, September 1990.

2. Proctor, F. H.: *The Terminal Area Simulation System. Volume I: Theoretical Formulation.* NASA CR-4046, April 1987.

3. Proctor, F. H.: *The Terminal Area Simulation System. Volume II: Verification Cases.* NASA CR-4047, April 1987.

4. Proctor, F. H.; Bowles, R. L.: Investigation of the Denver 11 July 1988 Microburst Storm with the Three-Dimensional NASA-Langley Windshear Model. *Windshear Case Study: Denver, Colorado, July 11, 1988,* DOT/FAA/DS-89/19, 1989.

5. Oseguera, Rosa M.; Bowles, Roland L.: *A Simple, Analytic 3-Dimensional Downburst Model Based on Boundary Layer Stagnation Flow.* NASA TM-100632, July 1988.

6. Vicroy, Dan D.: *A Simple, Analytical, Axisymmetric Microburst Model for Downdraft Estimation.* NASA TM-104053, DOT/FAA/RD-91/10, February 1991.

7. Ivan, Michael: A Ring-Vortex Downburst Model for Flight Simulations. *Journal of Aircraft,* Vol. 23, No. 3, pp 232–236, March 1986.

8. Schultz, Thomas A.: A Multiple Vortex Ring Model of the DFW Microburst. *Journal of Aircraft,* Vol. 27, No. 2, pp 163–168, February 1990.

9. Zhao, Y.; Bryson, A.: *A Simplified Ring-Vortex Downburst Model.* AIAA 90-0580, January 1990.

MONTE CARLO SIMULATION OF LAUNCHSITE WINDS
AT KENNEDY SPACE CENTER

Eric M. Queen*
Spacecraft Controls Branch
NASA Langley Research Center

Michael S. Warner**
National Science Foundation Fellow
George Washington University
NASA Langley Research Center

Daniel D. Moerder†
Spacecraft Controls Branch
NASA Langley Research Center

Abstract

This paper develops and validates an easily implemented model for simulating random horizontal wind profiles over the Kennedy Space Center (KSC) at Cape Canaveral, Florida. The model is intended for use in Monte Carlo launch vehicle simulations of the type employed in mission planning, where the large number of profiles needed for statistical fidelity of such simulation experiments makes the use of actual wind measurements impractical. The model is based on measurements made at KSC and represents vertical correlations by a decaying exponential model which is parameterized via least-squares parameter optimization against the sample data. The validity of the model is evaluated by comparing two Monte Carlo simulations of an asymmetric, heavy-lift launch vehicle. In the first simulation, the measured wind profiles are used, while in the second, the wind profiles are generated using the stochastic model. The simulations indicate that the use of either the measured or simulated wind field results in similar launch vehicle performance.

I. Introduction

The need for means of generating synthetic winds motivated a number of studies during the 1960's[1,2] which resulted in simulation models in which wind variation was represented via analog transfer functions[1], and by multistep regression models[2]. More recently, modelling of wind profiles was treated as a demonstration of a low order stochastic realization scheme[3]. This approach leads to multistep autoregressive models whose covariance satisfies an error criterion against the covariance of the actual measured data.

Another approach to the synthesis of wind profiles is used in the Global Random Atmosphere Model (GRAM)[4,5,6]. GRAM is a world-wide database of point statistics for northerly and easterly winds, in addition to atmospheric density, temperature, and pressure. Sample profiles are realized from the database via a simple first-order Markov perturbation model, under an assumption that the processes are Gaussian. The Markov model is formulated so that the pointwise means, variances, and correlations are reproduced exactly from the database. Correlations between parameters at different altitudes and locations are assumed to decay exponentially with distance.

In the next section of the paper, a collection of wind data from the KSC launch site is statistically characterized and used to develop a simple one-step wind simulation model in which northerly and easterly winds are assumed to be correlated Gaussian processes. The validity of this assumption is evaluated by statistical goodness-of-fit testing. The model represents vertical correlations by a decaying exponential model which is parameterized via least-squares parameter optimization against the sample data. Validation of the model is discussed in the section III. Since the motivation for this work is the construction of a model for use in launch vehicle Monte Carlo simulations, the validity of the synthetic model is evaluated by comparing two Monte Carlo simulations of an asymmetric, heavy-lift launch vehicle. In the first simulation, the measured wind profiles are used, while in the second, the wind profiles are generated using the stochastic model. Conclusions are presented in the section IV of the paper.

II. Model Development

This section describes development of the wind simulation model. The wind measurement data on which the model is based were provided to the authors courtesy of the Marshall Space Flight Center (MSFC). These data consist of horizontal wind velocities and azimuths tabulated on altitude, for 450 Jimsphere profiles measured at KSC between 1964 and 1972. Jimspheres are metallic coated balloons which are used to measure winds by tracking them with radar after release at ground level. This approach to measurement assumes that the balloon is entrained in the air mass, and that the air mass velocity is constant at each altitude, along the groundtrack flown by the balloon. All of the Jimsphere profiles were taken during the months of December through February. These months were selected for study because the winter wind environment at KSC is most difficult for launch operations.

For each month of data, 150 profiles were supplied, with data represented at vertical intervals of 25 meters from ground level up to 20 km. A number of the profiles were missing data either near the ground, or near the top of the profile. Figure 1 displays the number of samples available in the winter data set as a function of altitude.

As stated in the Introduction, the simulation model to be described below assumes a jointly Gaussian distribution for the northerly and easterly wind components. Therefore, the velocity/azimuth data were resolved into its northerly and easterly (N/E) components for the subsequent analysis. The sample means and standard deviations of these components are shown in figures 2 and 3. The cross-correlation between north-south winds and east-west winds was also investigated. As can be seen in figure 4 this cross-correlation can be significant at certain altitudes. The first step in this analysis is verification of the assumption that the wind data are jointly Gaussian at each altitude. This was done via goodness-of-fit testing[7]. Statistical goodness-of-fit testing is done by constructing a measure of the error between the sample statistics of a given population of data and those of the model distribution. This measure, itself a random variable, is chosen in such a manner that its probability distribution is known. The "significance" of the goodness-of-fit, then, is expressed by the probability that, given the assumption that the sample population did indeed come from the model distribution, a larger error could have been observed. In other words, it is the probability that the test user would be wrong if he or she concluded that the sample population was not drawn from the model distribution.

The assumption that the N/E wind data are jointly Gaussian was tested via two-dimensional chi-square tests at each altitude. The results of these tests is shown as figure 5. The lower graph in the figure is the number of degrees of freedom for the chi-square random variable which describes the fitting error. The chi-square test is based on frequencies in binned data, and the degrees of freedom are equal to the number of bins minus 6, at each altitude. Examination of the significance plot indicates that this was a rather noisy test, and that the average value of the fit significance was on the order of 30 percent. Based on traditional interpretations these tests[7], this does not indicate a particularly bad fit. Higher significance values, however, would provide stronger confidence in the model. One possible explanation for the spread in significance values stems from processing which the wind data underwent before it became available to the authors. These data are measured in velocity/azimuth format by radar at KSC at altitude intervals between 25 meters and 150 meters, depending on altitude and conditions. The profiles are then smoothed by a combination of automated and manual means, before being stored at 25-meter increments and delivered to MSFC. It is possible that this processing introduces effects into the

data which, upon performing the nonlinear transformation from velocity/azimuth into N/E coordinates, results in the small non-Gaussian effects seen in figure 5. Figure 6 displays significances for the separate fits of the N/E winds to their assumed marginal Gaussian distributions. These significances were obtained using the Kolmogorov-Smirnov test[7], which is applicable to continuous univariate random variables. It can be seen that these marginal fits are significantly cleaner and better than the joint fit. This observation is not inconsistent with the proposed explanation for the distortion in the joint distribution. Therefore, given the potential for uncertainty due to data smoothing and the relatively small (≤ 450 samples) sample size, the results displayed in figures 5 and 6 suggest that the Gaussian model is an adequate representation of the wind process.

Given the assumption that the wind data are Gaussian, simulated wind profiles can be expressed in the form of the sample mean plus a random perturbation term

$$\hat{W}_K(h) = \mu_K(h) + \delta_K(h) \qquad (1)$$

where $\hat{W}_K(h)$ is the simulated wind at altitude h, and the subscript K denotes either east or north, as appropriate. The mean wind at altitude h is denoted $\mu_K(h)$

$$\mu_K(h) = E[W_K(h)] \qquad (2)$$

where $W_K(h)$ is the K-component of measured wind at altitude h, and E is the expectation operator. A first order Markov process was chosen to model the wind perturbations, $\delta_K(h)$. This has a relatively simple mathematical form and it allows a free choice of the altitude separation between updates. The $\delta_E(h)$ and $\delta_N(h)$ are correlated to reflect the cross-correlations between east and north winds.

$$\begin{bmatrix} \delta_E(h) \\ \delta_N(h) \end{bmatrix} = \begin{bmatrix} \sigma_E & 0 \\ R_\nu(h)\sigma_N & \sigma_N\sqrt{1-R^2(h)} \end{bmatrix} \begin{bmatrix} \nu_E(h) \\ \nu_N(h) \end{bmatrix} \qquad (3)$$

Here $\sigma_K(h)$ is the standard deviation of the measured wind speed,

$$\sigma_K^2(h) = E\{[W_K(h) - \mu_K(h)]^2]\} \qquad (4)$$

and $R_\nu(h)$ represents the cross-correlation between the Markov processes ν_E and ν_N. This correlation is chosen to be

$$R_\nu(h) = \frac{R_N(h) - R_{NE}(h)R_{NE}(h-\Delta h)R_E(h)}{\sqrt{1-R(h)}\sqrt{1-R(h-\Delta h)}} \qquad (5)$$

Here R_{NE} is the measured wind cross-correlation coefficient,

$$R_{NE}(h) = E\{[W_E(h) - \mu_E(h)][W_N(h) - \mu_N(h)]\} \qquad (6)$$

This choice of R_ν insures that the cross-correlation between δ_E and δ_N will preserve the cross-correlation observed between the east and north wind data. The first-order Markov process, $\nu_K(h)$, is propagated as

$$\nu_K(h) = a_K(h, h-\Delta h)\cdot\nu_K(h-\Delta h) + \sqrt{1-a(h, h-\Delta h)^2}\cdot g_K(h) \qquad (7)$$

where $g_K(h)$ is taken from a Gaussian random sequence. The function $a_K(h_1, h_2)$ must obey

$$a_K(h, h) = 1 \qquad (8)$$

and

$$0 \leq a_K(h_1, h_2) < 1 \qquad (9)$$

but is otherwise free to be chosen to best model the measured data.

In the simulation model, the function $a(h, h - \Delta h)$ corresponds to the correlation coefficient between winds at altitude h and $h - \Delta h$ in the K direction. Two choices for the form of $a(h, h - \Delta h)$ were investigated. The first is a simple exponential model:

$$a(h, h - \Delta h) = e^{-\Delta h/b(h)} \qquad (10)$$

where $b(h)$ is a scaling function to be fit over the available data. The other model form is an exponential with a constant offset:

$$a(h, h - \Delta h) = [1 - c(h)]e^{-\Delta h/b_1(h)} + c(h) \qquad (11)$$

with $b_1(h)$ and mixing factor $c(h)$ to be fit to the data over a range of Δh. The range over which Δh varied in selecting the parameters

in (10) and (11) was selected as the interval from 0 to 2.5 km. This 2.5 km "window" provides a compromise between applicability of the model for large values of Δh and applicability of the model throughout the maximum number of altitudes available in the data. The process of determining these parameters is discussed in more detail in the appendix.

In order to reduce the amount of tabular data necessary to implement the model, the effect of calculating the model parameters in (10) and (11) for binned data was considered. By binning the wind data into 200-meter sections, the number of model parameters can be reduced by a factor of 8. Figure 7 compares the scale distances $b(h)$ for north and east winds for the single parameter model (10) using binned and unbinned data. The second model parameters for the binned data are shown in figure 8.

The goodness-of-fit of the vertical correlation model to the actual data was evaluated by examining the error between the model and data as a function of h and Δh. The error surface for the simple exponential model for east and north winds is shown in figure 9. The error surfaces for the second model are shown in figure 10. It is clear that the exponential with offset model gives a better fit, especially at low altitudes. The difference between the two error surfaces for each of east and north is shown in figure 11. The differences are small, suggesting that increased fidelity of the exponential with offset model is slight.

III. Model Validation

The wind model was tested by applying it to the simulation of an Advanced Launch System (ALS) class vehicle. The wind model is considered to be valid if vehicle simulations through the modelled wind exhibit behavior which is similar to the behavior of simulations through the measured wind. To test the accuracy of the wind model, an experiment was performed which compared the ensemble performance of a large number of launch trajectories of the ALS through the measured winds with the ensemble performance of the same number of runs through simulated winds. This section describes the vehicle model used, the issues important to validation of the model, and the results of the experiment.

The vehicle model is based on that of Pamadi & Dutton[8], an asymmetric, single booster configuration. The three degree-of- freedom simulation of this model uses winds input as tabular functions of altitude. All simulations were ascent trajectories run to 16.5 km altitude, since that is the highest altitude for which sufficient wind measurements were available. Staging was not an issue since it occurs at 153 seconds and 20 km altitude was usually reached after about 77 seconds. The vehicle was controlled with a pitch rate history, which was calculated without considering winds.

In the validation of a statistical wind model for launch simulations, the values of maximum dynamic pressure, q_{max}, the maximum product of dynamic pressure and angle of attack, $|q\alpha|_{max}$, and the maximum product of dynamic pressure and sideslip angle, $|q\beta|_{max}$ are important measures of the behavior of the simulation. These quantities are directly related to the loads experienced by the vehicle, and they are largely determined by the winds encountered. Other quantities of interest were the cumulative products of dynamic pressure with angle of attack and sideslip, $\int(q\alpha)^2$ and $\int(q\beta)^2$. Persistent differences between the magnitudes of the measured and simulated wind fields would manifest themselves in the relative values of these quantities.

The first Monte Carlo simulation consisted of 450 trajectories which incorporated the measured wind profiles used to determine the wind model parameters. Another 450 trajectories were simulated using wind profiles generated by the wind model. From each of these 900 trajectories, the values of q_{max}, $|q\alpha|_{max}$, $|q\beta|_{max}$, the times at which they occurred, and $\int(q\alpha)^2$ and $\int(q\beta)^2$ were recorded.

The means and standard deviations of these quantities, along with the 95% confidence intervals, based on 450 samples, are presented in figures 12 and 13. Figure 12 shows the mean final velocity, the mean final time, the mean q_{max}, the mean time of q_{max}, mean $|q\alpha|_{max}$, mean time of $|q\alpha|_{max}$, mean $|q\beta|_{max}$, mean time of $|q\beta|_{max}$, mean $\int(q\alpha)^2$ and mean $\int(q\beta)^2$ for each of the simulated and measured wind profile Monte Carlo runs. For all but $|q\alpha|_{max}$ and time of $|q\alpha|_{max}$ the error bars overlap, indicating very good agreement between the two methods. The difference in $|q\alpha|_{max}$ is probably due to non-Gaussian aspects of the measured data.

IV. Concluding Remarks

A KSC-specific launchsite wind model has been developed and tested. This model is based on wind measurements made at KSC in the winter months over a period of several years. These measurements were statistically examined. The wind model is propagated via a correlated pair of simple Markov processes. Binning of the model statistics reduced the amount of data needed to use the model with negligible effects on the results.

This model was validated by comparing the results of two Monte Carlo simulations, one using simulated winds and the other using measured winds. The results of these simulations were within 95% confidence intervals for many important quantities of interest. This indicates that the modelled winds are useful for Monte Carlo simulations of launch vehicles.

References

1. E. Gott, "Linear Regression of Interlevel Wind Velocities," IEEE Trans. Geoscience Electronics, Vol. GE-7, No. 1, Jan. 1969, pp. 44-47.

2. Bailey, J., Palmer, J., and R. Wheeler, "Launch Vehicle and Turbulence Response by Nonstationary Statistical Methods," NASA CR-846, Aug. 1967.

3. Menga, G., and N. Sundarajan, "Stochastic Modeling of Mean-Wind Profiles for In-Flight Wind Estimation - A New Approach to Lower Order Stochastic Realization Schemes," IEEE Trans. on Automatic Control, Vol. $AC-27$, No. 3, June 1982, pp. 547-558.

4. Justus, C., Roper, R., Woodrum, A., and O. Smith, "Global Reference Atmosphere Model for Aerospace Applications," AIAA J. Spacecraft and Rockets, Vol. 12, 449-450.

5. Justus, C., Fletcher, G., Gramling, F., and W. Pace, "The NASA/MSFC Global Reference Atmospheric Model - MOD 3 (With Spherical Harmonic Wind Model)," NASA CR-3256, March 1980.

6. Justus, C., Alyea, F., and D. Cunnold, "GRAM-88: Improvements in the Perturbation Simulations of the Global Reference Atmospheric Model," NAG8-078, Georgia Tech Project G-35-685, Oct. 1987.

7. Guttman, I., Wilks, S., and Hunter, J., *Introductory Engineering Statistics*, Wiley, New York, 1971.

8. Pamadi, B.N., and Dutton, K.E., "An Aerodynamic Model for ALS Vehicle", NASA TM in preparation,1990.

Appendix A

This appendix describes the process of determining the vertical autocorrelation coefficients for the simulated northerly and easterly winds as introduced in section II.

Two structures for the autocorrelation coefficient model were investigated. In the first, the autocorrelation coefficient $r(h, \Delta h)$ is modelled as $\hat{r}(h, \Delta h)$, a decaying exponential with a scale distance b, defined as a function of altitude:

$$\hat{r}(h, \Delta h) = e^{-\Delta h / b(h)} \qquad (A-1)$$

Here, Δh is the altitude increment between measured wind data points.

A second model was investigated which introduced a second scale distance, $b_2(h)$, and a mixing factor, $c(h)$:

$$\hat{r}(h, \Delta h) = [1 - c(h)]e^{-\Delta h / b_1(h)} + c(h)e^{-\Delta h / b_2(h)} \qquad (A-2)$$

The additional parameters of the latter model were introduced to provide additional flexibility in achieving a close fit to the sample autocorrelation coefficients.

Values for the parameters for the two model structures were calculated at each altitude by minimizing $J(h)$, the squared deviation of the sample and modelled autocorrelation coefficients:

$$J(h) = \sum_{i=1}^{k} \left[r(h, \tau_i) - \hat{r}(h, \tau_i) \right]^2 \qquad (A-3)$$

$$\tau_i = i \Delta h, \tau_{max} = k \Delta h$$

The deviations are summed over a range of altitude separations, τ, such that

$$0 < \tau < 2.5 km \qquad (A-4)$$

as mentioned in section 2. The sample autocorrelation coefficient, r, is calculated as

$$r(h, \tau) = \frac{1}{s(h)s(h + \tau)} \frac{1}{N - 1} \sum_{j=1}^{N} w_j(h)w_j(h + \tau) - N\bar{w}(h)\bar{w}(h + \tau)$$
$$(A-5)$$

$$\bar{w}(h) = \frac{1}{N} \sum_{j=1}^{N} w_j(h) \qquad (A-6)$$

$$s^2(h) = \frac{1}{N - 1} \sum_{j=1}^{N} \left[w_j^2(h) - N\bar{w}(h) \right] \qquad (A-7)$$

where w_j is the east or north component of the measured wind in the jth data profile, and N is the total number of profiles.

For each model, optimal model parameters were found using a Quasi-Newton algorithm, with convergence criterion:

$$\frac{\| \nabla J \|}{J} \leq 10^{-4} \qquad (A-8)$$

where ∇J represents the gradient of J with respect to the free parameters.

In running the optimization algorithm using the second model (A-2), the autocorrelations at first violated the constraint that the autocorrelation be between zero and one. The problem was resolved by bounding the values of the mixing factor c and the second scale distance as follows:

$$0 < c < 1 \qquad (A-9)$$

$$b_1 < b_2 < 10000b_1 \qquad (A-10)$$

These inequality constraints were each implemented in the unconstrained minimization through use of Valentine transformations. These transformations yielded the following equality constraints in terms of two unconstrained model parameters, α_1 and α_2:

$$c = \sin^2(\alpha_1) \qquad (A-11)$$

$$b_2 = (1 + 100\sin^2(\alpha_2))^2 b_1 \qquad (A-12)$$

Using the modified optimization algorithm, approximate autocorrelation functions were calculated using model structures (A-1) and (A-2) at base altitudes up to 14 km. This altitude limit ensured that a sufficient number of reasonable wind data pairs were available for calculating autocorrelation coefficients in (A-3). Recall that the autocorrelations are modelled over an altitude separation of 2.5 km. Thus, from the highest base altitude, autocorrelations can be calculated up to 16.5 km, the highest altitude used in the ALS simulations discussed in section III.

The resulting model parameters are in the form of tabular functions at the 560 altitude points (up to 14 km in 25-meter increments). The effect of binning on the fidelity of the model was investigated in an effort to reduce this number. The original wind data were binned into 200-meter altitude increments, and corresponding autocorrelations and model parameters were calculated. The results comparing the binned and unbinned cases are discussed below.

The scale distances for the first model over the altitude range are given in figure A-1, and the corresponding error surfaces are presented in figures A-2. Note that binning the wind data reduces the high frequency content of the scale distance results, while the errors for the two cases are virtually identical.

For the second model, the optimal model parameters are given in figure A-3. At the middle and highest altitudes for the north winds, the decrease in the scale distances in the binned case is compensated for by a corresponding increase in mixing factor. Otherwise the results are virtually identical. The similarities in the error surfaces presented in figure A-4 confirm the equivalence of the two sets of results.

Therefore, when the autocorrelation function is based on binned as opposed to unbinned wind data, it has only one eighth the number of model parameters with an insignificant loss of accuracy. For this reason, binned results are used for further analysis in section II.

Similarly, note that in using either the binned or unbinned wind data, the second scale distance in general rode the maximizing con-

straint, effectively making the second term in (A-2) a constant equal to the mixing factor. For this reason, the model was simplified by neglecting the second exponential function, leaving the model form as:

$$\hat{r}(h, \Delta h) = (1 - c(h))e^{-\Delta h/b_1(h)} + c(h) \qquad (A-13)$$

This model form was used for the Monte Carlo simulations, as described in section III.
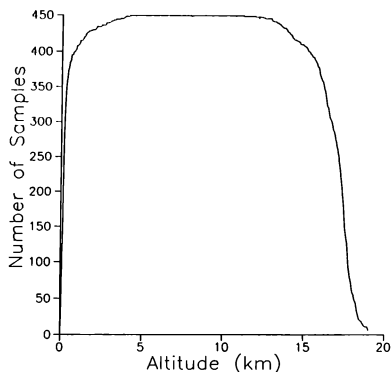
Figure 1. Number of Wind Data Samples at each Altitude

Figure 3. Mean and Standard Deviation for North Winds

Figure 2. Mean and Standard Deviation for East Winds
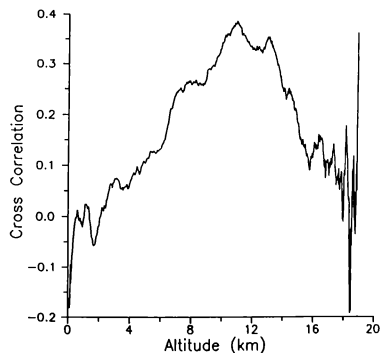
Figure 4. Cross Correlation Between East and North Winds with Altitude

313
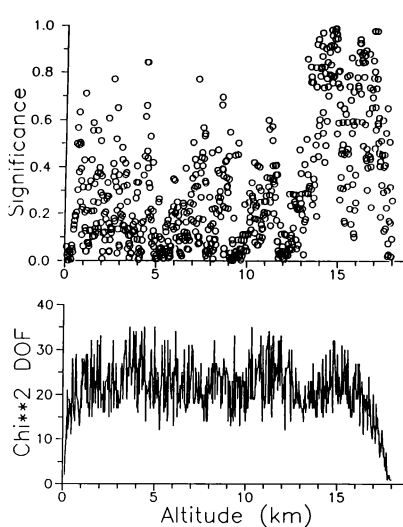
Figure 5. Results of Two-Dimensional Chi-Squared Tests on the North/East Wind Data

Figure 7. Exponential Scale Distances for First Correlation Coefficient Model for East and North Winds

Figure 6. Komolgorov-Smirnov Statistic with Altitude for East and North Winds
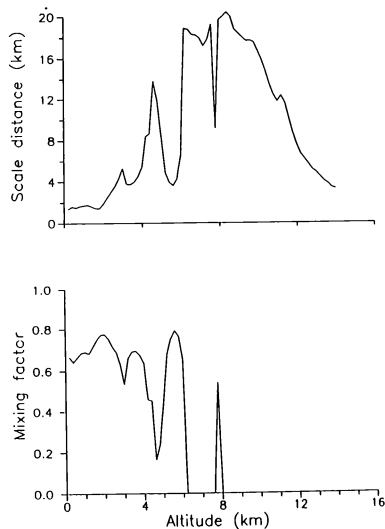
Figure 8a. Second Correlation Coefficient Model Parameters for East Winds
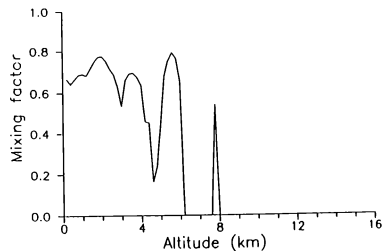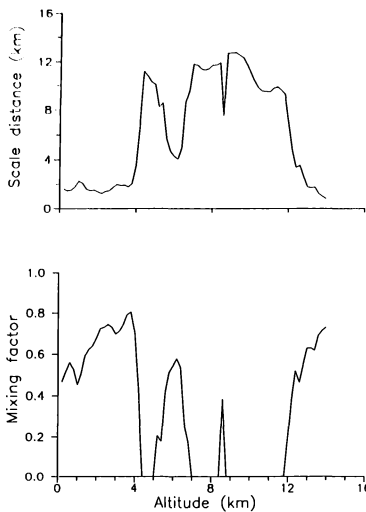
Figure 8b. Second Correlation Coefficent Model Parameters for North Winds
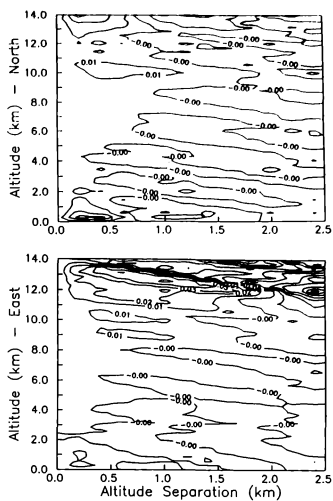
Figure 10. Second Correlation Coefficient Model Errors for East and North Winds
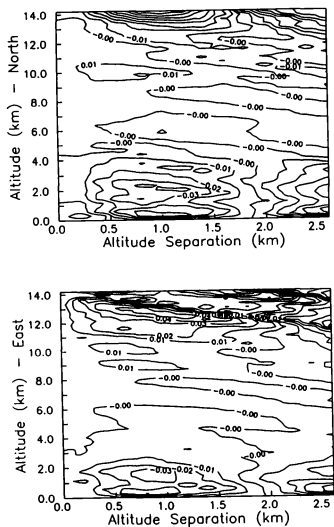
Figure 9. First Correlation Coefficient Model Errors for East and North Winds
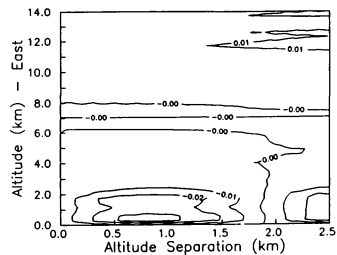
Figure 11. Differences Between Correlation Coefficient Model Errors for East and North Winds
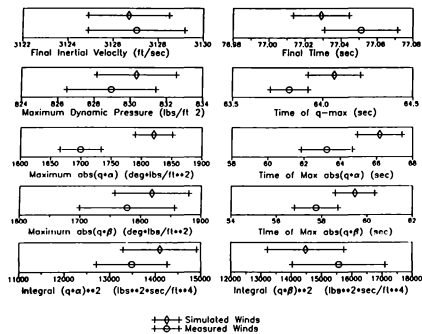
Figure 12. Comparison of Means of Statistics Using Measured and Simulated Winds
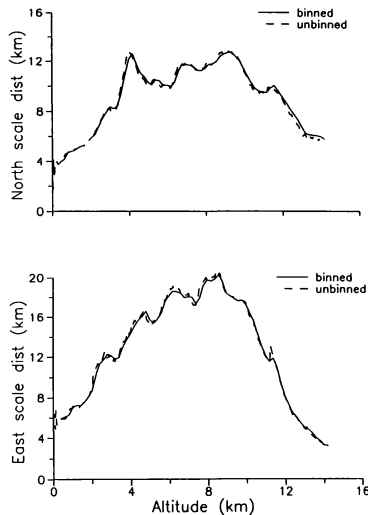
Figure A-1. Comparison of Scale Distances for Binned and Unbinned East and North Winds (First Correlation Coefficient Model)
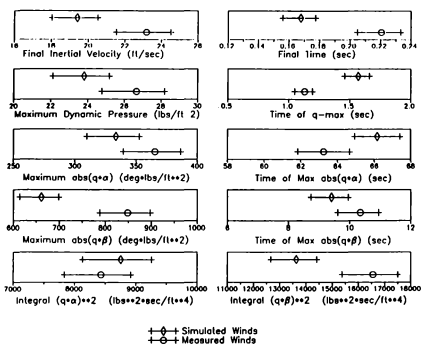
Figure 13. Comparison of Standard Deviations of Statistics Using Measured and Simulated Winds
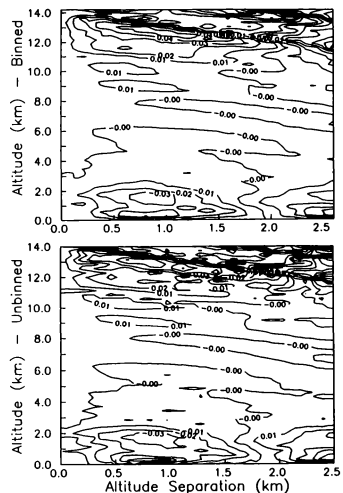
Figure A-2a. Comparison of Model Errors for Binned and Unbinned East Winds (First Correlation Coefficient Model)
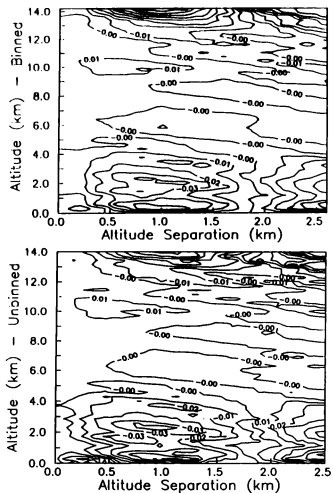
Figure A-2b. Comparison of Model Errors for Binned and Unbinned North Winds (First Correlation Coefficent Model)
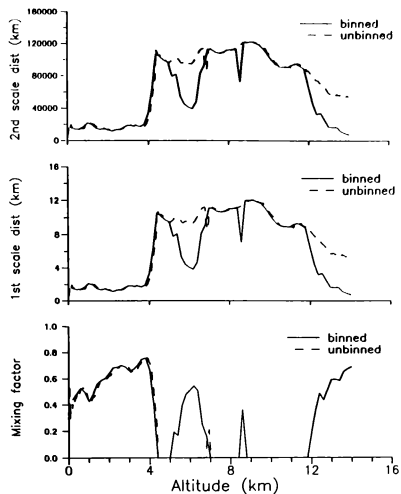
Figure A-3b. Comparison of Model Parameters for Binned and Unbinned North Winds (Second Correlation Coefficent Model)
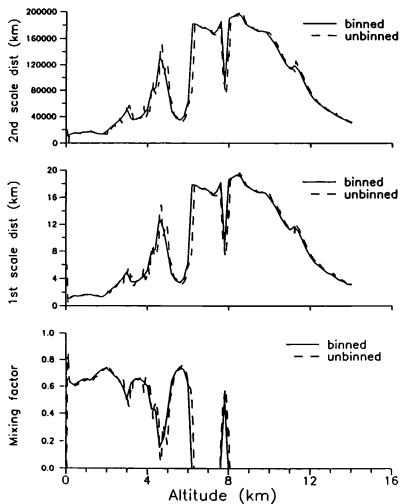
Figure A-3a. Comparison of Model Parameters for Binned and Unbinned East Winds (Second Correlation Coefficent Model)
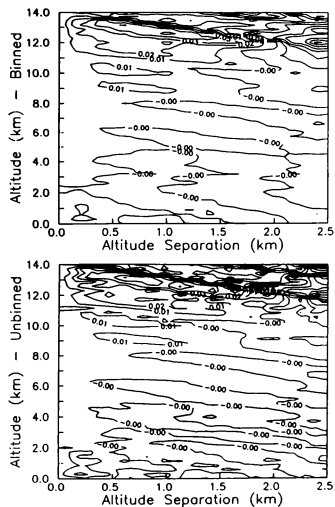
Figure A-4a. Comparison of Model Errors for Binned and Unbinned East Winds (Second Correlation Coefficent Model)
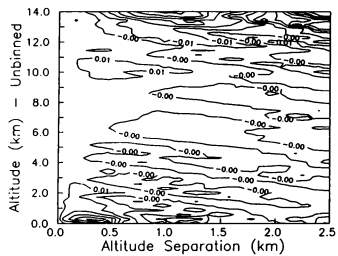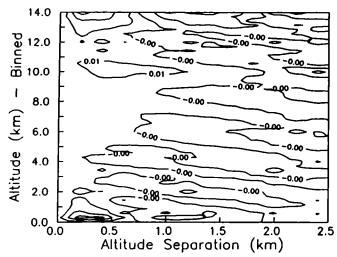
Figure A-4b. Comparison of Model Errors for Binned and Unbinned North Winds (Second Correlation Coefficent Model)

SYSTEM SIMULATION ON ADVANCED EARTH-ORBITAL SATELLITE SYSTEMS

Hwa-Ping Lee*
NASA Goddard Space Flight Center
Greenbelt, Maryland

Abstract

A system simulation capability for advanced Earth-orbital satellite systems is presented. It is based on analytical assessment capabilities formulated for generic satellite systems in Earth-orbital missions. The resulting **AEOSS** is a customarily tailored software coded within the framework of the Macintosh version of the Acius' relational database program, **4th Dimension**. It projects the required power, weight, and cost for a satellite system. These key elements are computed on the component and subsystem levels, and then the system level. Selected performance analyses for essential components and subsystems are included. Costs are assessed using the statistically determined cost models of the flown spacecraft that were categorized into classes in accordance with their functions and complexity. This software furnishes an option that permits known values of the parameters to be entered at all levels, and thus warrants results to be realistic and reliable. Information is of vital importance to advanced program planners and flight program managers.

I. Introduction

The system simulation for generic spacecraft systems that perform scientific observations and measurements in their Earth-orbital missions has been accomplished. Emphasis is placed on the estimates of the system requirements of the needed electric power, the weight, and the cost of a spacecraft, although capabilities of performance analyses for a number of major components and subsystems are also included. Information is essential to advanced program planners as well as to managers of flight programs. The developed analytical assessment capabilities were implemented in a relational database program *4th Dimension*[1], that is suitable for all objectives and applications. The use of a commercially available database software reduced the software development effort considerably, but exploited all inherited features of that computer program.

This work was a task of the R&D program entitled *Advanced Earth-Orbital Spacecraft Systems*. The initiative was to support the NASA's objective of *a leadership role in space technology for Earth-orbital free-flying spacecraft*. As NASA's efforts in the Earth-orbital satellites continue to focus on long-term research in areas such as astrophysics, Earth science, communications, etc., the trend in future satellites is toward physically larger and functionally more complex spacecraft systems than current ones. The expansion initiates in the instruments, which have increased steadily in size, complexity, and sensitivity[2,3].

In orbit, the electric power is virtually required by all onboard functional components and instruments. This subsystem commands a high-priority status in a spacecraft system.

The selected orbit in any mission has always been determined through a trade-off between coverage objectives and the capabilities and economics of rocket-booster selection. There are significant cost differences among various boosters (e.g., Scout, Delta, Atlas, Titan, etc.). The costs of commercial launch services for the expendable launch vehicles, as supplied by the NASA Headquarters, are given in Table-1. Capabilities of delivering the payload weight (defined as the total mass delivered to the target orbit) by different boosters

to a 100-NMi circular orbit are digested and shown in Table-2. For illustration, a Delta-II would cost approximately $4,600 per pound of payload to a 100-NMi circular orbit, and $5,100 per pound if an Atlas-I is selected instead.

The system weight of a spacecraft (the sum of all weights of the components, subsystems, and payloads) has direct bearing on the economics of launching a payload. For a given booster, any weight savings realized by reducing the weights of the composing components of a spacecraft may be used either to increase the payload weight, or to raise a given payload into a higher desirable altitude.

The costs of individual components, the subsystems and the system of a spacecraft are crucial, especially in the face of the budgetary constraints; the program planners and the managers are forever questing for more performance efficiency and cost effectiveness. The three key elements, the required electric power, the weight, and the cost, are dominant factors that are weighed in the decision-making process by the advanced program planners at NASA Headquarters and by the managers of flight programs, both on the subsystem level and the system level.

The knowledges of performances of major components and subsystems are essential in assessing the weight and the electric power requirements. Unless relevant data or empirical expressions pertinent to the components and subsystems are known, one has to resort to analytical results for estimations. The computed results of a spacecraft, based on this analytical assessment capability, may serve as benchmarks to gauge any performance enhancements or improvements when new or improved components substitute their counterparts in a similar spacecraft.

An Earth-orbital spacecraft system contains interwoven multidisciplinary technologies that can be categorized by the following six subsystems: (1) Electric power, (2) Thermal control, (3) Structure, (4) Auxiliary propulsion, (5) Attitude control, and (6) Communication, command, and data handling (CC&DH). The subsystems will be discussed individually in the following sections. An overview of the role played by each subsystem with specific components in a spacecraft will be given. The evolution of applied technologies and their features, if applicable, will be briefly reviewed. The design analyses or performance assessments of selected components and subsystems will be given. To keep this task manageable, only the state-of-the-art selected components are included. The equations that were assembled to evaluate the design adequacy and to estimate performances have been thoroughly scrutinized. Frequently, these equations needed to be further reduced, appropriate for the system analysis rather than dwelling on details at the component level.

With the exception of a few cases whose working equations are presented later for illustration, Reference 4 should be consulted for all discussions, details of analysis, equations and databases.

The cost models are statistically determined empirical expressions that were deduced from the flown spacecraft being categorized into classes according to their natures of mission, structural complexity, and weight ranges[5]. The computed costs are initially expressed in millions of the 1980 dollars; the inflation indices are then applied to the results of individual subsystems and the spacecraft system to project the costs in any year for a planned new satellite system.

----
* Member AIAA

The developed equations along with the established databases for various subsystems were computerized on a microcomputer environment for easy access and portability. For economic reason, it was coded with a relational database program, the Macintosh version of the Acius' **4th Dimension**. Such an approach, making the best use of the utility program's features, and thus reduces the coding efforts regarding the software infrastructure that would otherwise be required in programming.

The acronym of this customarily tailored database software is **AEOSS**, abbreviating for **Advanced Earth-Orbital Spacecraft Systems**[6]. AEOSS provides a skeletal structure and contains all essential materials delt in this study. It can not only perform some design analyses and evaluate performances for selected components and subsystems but also possesses features permitting specified values for parameters to be entered directly at all levels as an option. The software can be modified or expanded easily. The current version, however, is sufficient to provide information for all intended applications, and is capable of yielding results of the three key elements for the managerial decision-making process. By its own right, **AEOSS** is useful in performing preliminary design analysis and parametric studies. Examples will be presented for illustration.

II. Electric Power Subsystem

Electric power subsystem requires only two basic elements to function: An energy source, and an energy converter that converts the source energy to electrical energy. The power subsystem, according to the energy source, can be classified into three types: (1) Nuclear power, (2) Electrochemical, and (3) Solar-photovoltaic[7].

A variety of space nuclear power systems, both static and dynamic, have been studied and proposed for applications. Only static systems, however, have been flown with outstanding performance records and potentials in the future flights. Dynamic systems are not ready for flight in the foreseeable future because of the complexity in the rotary machinery, the related maintenance requirements, and concerns in the nuclear waste disposal problem, etc.[8]

Electrochemical systems that include batteries and fuel cells have been used only for missions of short duration. They are limited by the onboard stored energy storage capacity. Therefore, they will be treated merely as an auxiliary power storage unit in association with the photovoltaic system.

The solar-photovoltaic power system has been employed extensively for low Earth-orbiting (LEO) and geosynchronous (GEO) spacecraft[9,10]. It is the sole type using the renewable solar energy that is always present in space. It also has accumulated an extensive service record and is, therefore, the primary power subsystem to be considered in detail.

Spacecraft power requirements of the United States spacecraft started with less than 1 watt, then ranged upward to 1.5 kw as the typical requirement since the seventies (except the Skylab which had a power requirement of 6.5 kw). Approximately, 50% of the power was for the payload instruments and the other 50% was to meet the power demands for other subsystems. Currently, typical power densities of the electric power system range from 2 to 5 w/kg.

An Earth-orbiting spacecraft experiences the alternating sunlight and eclipse periods in orbits. The durations of these periods govern the exit temperature of a solar array from the eclipse and the solar array's operating temperature which, in

turn, regulates the output of power conversion. In addition, the effects of varying values of thermophysical properties of solar cells (including the surface emissivity, absorptivity, and the thermal capacitance) on the solar array operating temperature were studied. The sizing of a solar array with the required storage battery is presented in Fig. 1.

III. Thermal Control Subsystem

To survive the environments of extremely cold space and heating fluxes, temperature-sensitive elements in scientific payloads have to be maintained within specified temperature ranges for survival and operation. Minimal temperature gradients in structures must be kept when pointing accuracy is crucial. Proper thermal control may reduce or even eliminate thermally induced vibrations and fatigues in a cyclic heating environment. A variety of methods and devices for thermal control, in both active and passive categories, has been considered.

Applying thermal coatings to surfaces of a spacecraft is a simple yet effective method in controlling spacecraft temperatures. Experimentally verified data, in twelve tables, on emittance and absorptance of coatings have been collected[11].

Besides the multilayer insulations that are capable of isolating shielded structures from their surrounding thermal environment, the other passive thermal control method to reduce or eliminate temperature fluctuations is to store up available energy by utilizing the characteristics of the heat content of phase-change materials. Some phase-change heat-storage materials were selected[12].

A space radiator is a device that dumps waste heat for a spacecraft. It can be either a passive or an active thermal control apparatus depending on designs. A space radiator would contribute considerable weight to a thermal control sub-system[13-15]. The current technology of radiator at 20~30 w/kg must be significantly improved to keep the spacecraft weight within available booster capability for high-altitude orbits. Radiators must also be deployable for the anticipated high power levels. A collection of equations detailing the space radiator design is presented in Fig. 2.

Another thermal control device, the thermal louver, in effect, provides varying effective surface properties for a radiating surface[16]. It is commonly used in spacecraft that encounters cyclic thermal changes.

Heat pipes are employed to serve as high efficient thermal conductors. In addition to the basic constant conductance heat pipe, a variable conductance heat pipe has been developed and applied in space to provide a precise control of an isothermal operation when the source and sink thermal resistances were unequal[17,18].

Similar to the heat pipe, the capillary pumped loop (CPL) is a two-phase thermal conducting device capable of efficiently transferring heat with small temperature differentials and no external pumping power requirements[19]. The features have been demonstrated in the ability to transfer large heat load over long distance at a controlled temperature. Although this technology would never be used in a satellite because of the size, it was briefly discussed in view of its potential of this emerging technology.

Another active thermal control is to apply electric heaters to the body of spacecraft where a specified temperature level is required in an otherwise too low temperature field.

Although thermal contact resistances can hardly be classified as a means of thermal control, their effect at a joint is to restrict the heat flow and thus to influence temperatures. Some experimental data of the thermal contact resistance were collected for reference[20].

IV. Structure Subsystem

To provide a stable support and to ensure the structural integrity during all phases of the mission are the primary objectives of an adequate structural design. Structural members, therefore, are designed to withstand all anticipated loads, shocks, and vibrations as specified for the ground qualification and acceptance tests and ultimately for the launch loads. The launch loads consist of static and dynamic accelerations and acoustic noise as well. The load factors, which are the combined values of the steady-state and dynamic effects, are 3σ maximum values applicable at the center of gravity of the spacecraft structure in service. The stiffness levels of structure are required to keep the fundamental resonant frequencies of the spacecraft above certain levels in order to decouple the spacecraft's main resonance from the launch vehicle's dynamic excitations from all directions.

In design analysis, factors of safety are applied to the limit loads, pressures, and thermally induced stresses. Equally important in judging the results of design analysis is the margin of safety, which indicates the strength margin remaining in structure members.

Analytical solutions of stresses and vibration modes for a spacecraft normally depend on some mainframe structural analysis computer programs, such as the NASTRAN. The user may still require to manipulate the data of computer output for desired results that are useful in deciding acceptance or rejection of the structural design. The two yield criteria, namely, the VonMises yield criterion and the Tresca yield criterion, are commonly employed in judging results of the static analysis. As for results of vibration analysis, the direction of the nth-mode in a dynamic analysis is often an intuitive selection. When the configuration of spacecraft is complex in addition to irregular supporting points, an intuitive exercise would be very difficult and unreliable, especially in determination of higher modes. A methodology that determines the predominating directions of individual modes compatible with the NASTRAN outputs was developed.

There is a myriad of materials suitable for aerospace applications, and they have been well-documented. However, the trend is to use lighter and stiffer composite materials whose mechanical properties could be predetermined in the manufacturing process. The mechanical and thermophysical properties of frequently used materials, including metals and composite materials, were selected for quick reference.

V. Auxiliary Propulsion Subsystem

The auxiliary propulsion subsystem is to provide a controlled impulse for orienting the spacecraft spin axis, correcting the orbit, adjusting the attitude, etc. after the spacecraft has achieved its orbit. Over the life of a spacecraft mission, this subsystem is to make the orbital corrections many times in order to maintain the requirements for the parameters of the spacecraft.

The required velocity increments for correcting the orbital errors and to predict the required propellant and tankage, then their weights, are included[21]. Equations are shown in Fig. 3.

VI. Attitude Control Subsystem

Stabilization of an orbiting satellite is important to maximize the usefulness and effectiveness of the satellite. Some scientific payloads also need to point at certain directions in space accurately. The types of control include (1) Transient attitude steering or maneuvering to achieve various attitude, (2) Steady-state pointing in the presence of ambient disturbances, and (3) Maintaining attitude in the presence of onboard disturbance.

Whether the final orbit is a low Earth orbit (LEO) or a geosynchronous orbit (GEO), the process and control mechanism for attitude control of a satellite are similar. The sequence to achieve the final attitude of a satellite does depend on the orbit, the mode of launching, and the nature of the mission. The final attitude generally can be accomplished by the spin stabilization or three-axis stabilization. The equipped sensors and accessories onboard the satellite for the attitude determination and control are selected for the mission. Therefore, the size, weight, and required power of individual components may vary greatly, depending on the function to accomplish, the required accuracy, and the planned orbit for the satellite[22].

There are four major functional groups in the Attitude Control Subsystem: (1) Stabilizations for control, (2) Attitude measuring sensors, (3) Attitude actuators, and (4) Attitude control electronics.

The following types of stabilization have been employed in satellite attitude control: (1) Gravity-gradient stabilization, (2) Three-axis stabilization, (3) Spin stabilization, (4) Magnetic stabilization, and (5) Solar radiation stabilization. The first three are principal methods.

Designs of attitude measuring sensors vary in a broad spectrum, depending on the mission and accuracy requirements of the satellite. The categorized five types are: (1) Earth/Horizon Sensors, (2) Sun Sensors, (3) Star Sensors, (4) Magnetometers, and (5) Gyro-scopes (solid state rate transducer).

The attitude actuators are used to correct the attitude of a satellite in order to attain and maintain the desired orbit. Selections of type depend on the mission and payload requirements. Attitude actuators can be passive type or active type. The active types use expendables such as hydrazine gas. Some actuators are listed below:

- Momentum and reaction wheels or gyros
- Magnetic torquers
- Nutation dampers
- Thrusters or reaction control jets
- Scanwheels

The kernel of the Attitude Control Subsystem is the attitude control electronics. It provides the interface with attitude measuring sensors to obtain attitude data, also with actuators to command them to operate in a controlled manner, either to change the spacecraft attitude to the desired height or to maintain the spacecraft attitude in the desired one.

VII. CC & DH Subsystem

CC&DH subsystems are essential to maintain and operate an orbiting spacecraft in a desired orbit. Information on the satellite position and conditions is needed so that the proper commands can be sent from the ground to the satellite to achieve the goal. Telemetry, tracking, command, and

communication provide the means of monitoring and controlling the satellite operations[21].

Scientific satellites are instrumented with various sensors to measure their environmental parameters and scientific phenomena occurring in the vicinity of the spacecraft or far away in space. The sensors measure some physical parameters, such as pressure, temperature, current, voltage, acceleration, radiation, position, etc. The measured data of various parameters of interest must be processed (converting measured quantities into electrical signals) and telemetered by the telemetry system, that contains a device transmitting the measured data to the ground. For efficiency, multiplex (i.e., simultaneous transmission of several types of data) is necessary. Either the time-division multiplexing or the frequency-division multiplexing can be employed separately or in combination. In addition, the choice of either an analog system or a digital system is optional.

The total telemetry bandwidth requirements and telemetry multiplexing allocations are determined from the telemetry instrumentation and payloads schedule. Each set of data points is to be converted to an equivalent PCM (pulse code modulation) bit rate and the complete list of a schedule is summed up to dictate the required telemetry bandwidth. The total number of data points determines the size of the telemetry multiplexing.

Command is the means to send signals from the ground station to the satellite to establish the control of the satellite. Two types of command are available, they are: (1) The value commands - digital words of various lengths, and (2) The pulse commands - also called the descrete ON/OFF relay commands.

Communication is the system to link the satellite with the ground operator and to transmit the measured data or to send the command signals, to and fro, between the two terminals. Communications between an orbiting satellite and the ground operator generally use frequency modulation (FM) or pulse modulation (PM). The major factors in transmitter design that influences the input power requirements are the RF output power, operating frequency, data rate, and bandwidth. The output power is a function of the antenna gain and directionality, maximum range, and the ground station receiver characteristics. Transmitter power requirements increase with bandwidth and are a direct function of the data transmission rate, type of modulation, and stability requirements.

Receivers are installed in a satellite to receive and amplify the command signals sent from the ground station. The commands can be both real-time command and stored-time command.

In general, the uplink (ground to satellite) modulation should be simple for demodulation purposes. The onboard receiver should not be complex in consideration of the weight and reliability factors. The downlink (satellite to ground) modulation should require minimum power to transmit the information at a specified data rate, etc.

Tracking is the process of observing the motions and collecting data to plot the flight path of the satellite. The currently available tracking techniques are: (1) Optical tracking, (2) Radio tracking, (3) Radar tracking, and (4) Infrared tracking.

For transmitting and receiving radio frequency signals, antennas are designed and equipped to match the proper impedance of the transmitter or receiver in order to exchange the maximum RF energy. The basic and simplest antenna is the isotropic radiator, virtually a rod, which radiates RF energy uniformly in all directions. Antennas with concentrators or reflectors, having the following types of (1) Paraboloidal reflectors, (2) Cassegrain aerials, (3) Open Cassegrain types, (4) Horn reflectors, and (5) Steerable horn antennas have been designed to enhance the strength of the transmitted signal by concentrating the RF energy into a narrow beam.

VIII. Cost Estimation

In estimating costs of future spacecraft, the cost models adopted herein are based on that of the spacecraft subsystems that were originally developed by PRC Service Co.[5], and employed by the GSFC Resource Analysis Office. Any improvement of a component or a new addition to a subsystem would have impact on the overall cost. The actual cost, if known, should be entered directly into the cost estimations for the system.

The subsystems and system cost models were developed by using the database of a series of protoflight and follow-on flight unit spacecraft subsystems. The cost estimate is performed by (1) Estimating the appropriate set of subsystem cost estimating relationships, (2) Estimating the appropriate spacecraft system level cost estimating relationships, (3) Summing the resultant costs obtained in the preceding two steps to obtain total spacecraft costs.

Table-3 identifies the spacecraft subsystem and system level elements incorporated in the cost model. Cost participation in the individual system level is provided by applying the functional activity percentage factors to the overall system level cost estimates.

The cost models and factors were derived on the basis of 13 historical spacecraft that span a period in the launch year from 1970 to 1981. In addition to launch date, these spacecraft were diverse in terms of mission objectives, size, complexity, and degree of inheritance.

The computed costs are in millions of 1980 dollars, exclusive of prime contractor's fee. For estimating the cost of a new start, the basic cost model has to be adjusted to reflect the dollars in an appropriate year by applying the escalation factors, which is also known as the new start inflation index. It is issued and revised periodically by NASA Headquarters.

The cost model database was limited to earth-orbital, unmanned spacecraft with a mission other than military communications.

The following assumptions and constraints were imposed in deriving the cost models. Wherever the actual cost of an item is known, modifications should be made in the costing applications.

- The cost model database is limited to conventional expandable rocket-launched, unmanned, earth-orbital spacecraft, excluding any military communications satellites.

- The cost model provides protoflight and follow-on unit cost estimates at the subsystem and system levels in millions of 1980 dollars.

- The protoflight cost reflects both nonrecurring and recurring situations. Follow-on unit cost is limited to the recurring costs.

- The following costs were excluded in the formulation:

- All costs for in-house civil service manpower
- Prototype subsystems
- Experiments (except for communication equipment)
- Integration of experiments to the spacecraft
- Costs of ground support equipment beyond the first set
- Spare subsystems
- Launch and mission operations
- Propellant

• Early test models or mock-ups are included in the appropriate subsystem nonrecurring costs. Normal allowances of spare components are reflected in the subsystem recurring costs.

• Software developed in support of specific flight subsystems are included in the costs of those subsystems. Software developed for ground support equipmentused in testing and integrating the spacecraft are included in the ground support equipment costs. All other software costs are excluded.

• All included software costs are considered a non-recurring.

• Costs and technical data for items provided as government furnished equipment have been incorporated, wherever possible, into the appropriate subsystems.

• Each cost model cost estimating relationships should be used with caution for application beyond the range of values for the cost explanatory variable.

• Customized cost estimates should be used in lieu of estimates provided by the cost model where circumstances and available data suggest that improved estimates will result.

To begin costing, the required inputs include the number of spacecraft, the weights for all subsystems, the type of spacecraft (Non-Explorer or Explorer), and the complexity of the spacecraft (Complex or Simple).

For single spacecraft programs, only the protoflight segment of the cost model is to be exercised. For multi-spacecraft programs, both the protoflight and the follow-on unit portions have to be computed.

The cost models for all subsystems have a general form of

$$C_i = a_i W_i^{b_i} \qquad (1)$$

where

C = cost in millions
W = weight in lbs
a = coefficient in the cost model
b = exponent to the weight in the cost model
i = subscript denoting the ith subsystem

The cost model for the spacecraft system has the form similar to Eq. (1) with uniquely defined variables, i.e.,

$$C_{s,80} = a_s K^{b_s} \qquad (2)$$

where

$$K = \sum_i C_i \qquad (3)$$

is the sum of cost of the subsystems and the subscript s signifies the spacecraft system.

A summary of the spacecraft subsystem and system level cost estimating relationships is presented in Table-5. The results so obtained give protoflight and follow-on unit cost estimates at the subsystem and system levels. However, all costs so calculated are in millions of 1980 dollars, and they are exclusive of fee. The current total cost of a spacecraft system may be estimated by the following modification:

$$C_{s,current} = f_{inf} \cdot C_{s,80} + F_{pc} \qquad (4)$$

where

f_{inf} = inflation factor
F_{pc} = prime contractor's fee, generally 10% of the current cost of the spacecraft system
$C_{s,time}$ = the time indexing the current and the 1980 costs of a spacecraft system.

If the actual cost of a cost element is known, whether a component or a subsystem, the known value should be used instead of the calculated result. In addition, if the known cost in a year (denoted as the reference year) is other than the year of 1980, that reference cost has to be converted to the cost base year of 1980 first, then adjusted to the current year through the use of the appropriate inflation factors of those two years, i.e.,

$$C_{i,current} = \frac{f_{inf,current}}{f_{inf,reference}} C_{i,reference} \qquad (5)$$

where

$C_{j,time}$ = known or calculated cost at subsystem or system level with reference to time (the current or the reference year)
$F_{inf,time}$ = the inflation factor to adjust between 1980 and the time of interest.

To simplify cost estimating applications, two special worksheets are prepared and presented in Figs. 4 and 5. The upper lines of each worksheet provides for identification of the project as well as the generic costing information. The lower portion provides step-by-step procedures for cost estimation of the subsystems, (the constants in the cost models are already given in Table-4) and of the system where the spacecraft system level functional percentage factors are explicitly incorporated with the associated activities.

IX. Software Implementation

The developed equations in the preceding sections have been coded into a computer program that was to operate on a microcomputer for easy access and portability. A relational database program, the Macintosh version of the Acius' *4th Dimension*, was selected to implement the materials. The selection of a utility program was prompted by the economic consideration. Such an adaptation, exploiting the utility program's features, and thus reduces the coding efforts regarding the construction of the software infrastructure that would be otherwise required in programming.

This feature-ladden relational database software meets all requirements for selection. The criteria are: (1) To provide needed mathematic operations as an underlying processing engine, (2) To function as a normal database management program, (3) To be user-friendly in applications, and (4) To permit future expansion without overhauling the program structure.

This customarily tailored database software has been named *AEOSS*, abbreviating for the *Advanced Earth-Orbital*

Spacecraft Systems. The *AEOSS* provides a skeletal structure that can easily be expanded and modified. This version, however, is sufficient to perform normal applications, and is capable of yielding results for the three key elements in the managerial decision-making process.

There are three environments provided by the software for applications, namely: (1) *Design*, (2) *User*, and (3) *Custom* environment.

The *Design* environment was employed to create the custom application program *AEOSS*. This environment allows one to create or alter the file structure, add or delete fields of files, design new or modify layouts, and write procedures for the associated layouts.

The inherited structure of *AEOSS* is composed of several files: The main files, the subfiles of the main files, or the linked files that are linked through a related field. Each file of any type is composed of fields. Each set of these fields constitutes a record. The subfiles work in much the same way as the files but are composed of subrecords.

The user may enter, view, and modify data through templates called layouts. There are two types of layouts: The input layouts and the output layouts. Each layout may associate with a layout procedure that is a series of programming statements written in *4th Dimension's* special language, and this procedure executes each time its associated layout is active on screen.

In application, the user enters *AEOSS* into a Macintosh, which has already been loaded with *4th Dimension*. After entering the passwords that select one of the three environments, a menubar will display at the top of the screen.

For the selected *Design* environment, the appeared main menubar, Menubar #1, has the form as shown in Fig. 6, which also displays the contents under each menu. Similarly, if the *User* environment or the *Custom* environment is selected, the menubar, Menubar #2 or Menubar #3, as shown in Figs. 7 and 8, respectively, will be displayed.

Under the Subsystems menu, there are six defined subsystems: Electric Power, Thermal Control, Structure, Auxiliary Propulsion, Attitude Control, and CC&DH. Each subsystem selected under this menu will call up a new menubar of its own. The six menubars are replicated in Fig. 9 for the individual subsystems.

The Subsystems menu differs from the Subsystem Data menu in that the former includes performance analyses of individual components and some defined subsystems that require the user to enter data for various parameters unique to the cases, and the latter, on the other hand, requires direct user entries of the summarized data, which can be calculated or specified values, for the listed parameters of all subsystems and their components as well. These parameters include weight, size, power consumption, and temperature.

The user first enters the system information through the System Info menu, then enters data into the subsystems of concern under the Subsystems menu for detailed analyses, and finally enters the calculated or specified values through the Subsystem Data menu to input data for computing the required end parameters. The summary results will be channeled automatically through the Summary of S/C System under the Formats menu.

The Formats menu allows the user to select different layouts to be viewed or printed. The contents are Summary of S/C system, Cost models, and the Spacecraft System totals.

The Tables menu contains Cost Constants, the NASA Inflation Index, Physical Constants, Weight Limits, and Modify Table. The last one is a special provision permitting the user in the *Custom* environment to expand or modify data in some tables without altering any layout procedures.

X. Illustration

Because of limited space, only a few examples exhibiting the results of performance analysis at the component level and the subsystem level were selected for demonstration. The three illustrative problems are those corresponding to the three formulas sheets given in Figs. 1, 2 and 3. The individual input variables and the calculated results are fully self-explanatory, and they are shown in Figs. 10, 11 and 12, respectively.

The process of cost estimating has been presented in Figs. 4 and 5 for the protoflight spacecraft and the follow-on unit(s), respectively. The use of the spacecraft subsystem cost model to generate the cost estimate for a hypothetical flight project ABC is presented. This hypothetical flight project consists of a protoflight spacecraft and three follow-on units. The results demonstrate to the point of adding fees (such as the prime contractor's fee, etc.), as necessary.

The costs shown in Figs. 4 and 5, respectively for the protoflight spacecraft and the follow-on units, project the final costs in terms of millions of 1980 dollars first, then are revised to reflect the amount in the planned year of 1991 by multiplying the inflation factor of 1.884. The total project costs are the sum of the cost for a protoflight unit and the costs of three follow-on units in the planned year.

XI Conclusions

A system simulation capability for generic advanced Earth-orbital satellite systems has been presented. The resulting equations and databases have been coded into a micro-computer program named **AEOSS**, that is a customarily tailored software coded within the framework of the Macintosh version of the Acius' relational database program, *4th Dimension*.

AEOSS contains capabilities of performance analyses for selected essential components and subsystems, and is capable of projecting the required power, weight, and cost for an Earth-orbital satellite system.

Costs are estimated using the statistically determined cost models of the flown spacecraft that were categorized into classes in accordance with their functions and complexity. This software permits known values of the parameters to be entered at all levels as an option, that warrants the results to be realistic and reliable. The information is of vital importance to advanced program planners and flight program managers at all levels.

AEOSS is versatile, it is useful also in performing preliminary design analysis as well as parametric studies.

References

1. Walden, J.,"4th Dimension User's Guide", Aclus Inc., 1987.

2. Studer, P. A.,"Generic Spacecraft Systems Technology", NASA-GSFC, NASA TM-86135, Aug. 1984.

3. Hudson, W. R. & Pawlick, E. V.,"Spacecraft Technology Requirements for Future NASA Missions", AIAA Paper-1160-CP, AIAA Space Systems Technology Conference, June, 1986.

4. Lee, H. P.,"System Analysis on Advanced Earth-Orbital Spacecraft Systems", NASA-GSFC, to be published.

5. PRC Systems Services,"Spacecraft Subsystems Cost Model", Huntsville, AL, A contract report, Jan. 1983.

6. Lee, H. P.,"AEOSS Design Guide for System Analysis on Advanced Earth-Orbital Spacecraft Systems", NASA-GSFC, NASA TM-100772, Dec. 1990.

7. Barthelemy, R., et al,"Aerospace Power Systems - A Building Surge", AIAA Aeronatics & Astronautics, Feb. 1979.

8. English, R.E.,"Power Generation from Nuclear Reactors in Aerospace Applications", NASA TM-83342, 1982.

9. Goldsmith, P. & Reppucci, G.,"Advanced Photovoltaic Power Systems", AIAA Paper-77-506, Mar. 1977.

10. Slifer, L.W. & Billerbeck, W.J.,"Synchronous Orbit Power Technology Needs", NASA-GSFC, NASA TM-80280, Apr. 1979.

11. Henninger, J.H.,"Solar Absorptance and Thermal Emittance of Some Common Spacecraft Thermal-Control Coatings", NASA Reference Publication-1121, Apr. 1984.

12. Tye, R.P., et al,"Thermal Energy Storage Material Thermophysical Property Measurement and Heat Transfer Impact", NASAS CR-135098, 1976.

13. Mackay, D.B.,"Design of Space Powerplants", 1st ed., Prentice-Hall Inc., 1963.

14. Russell, L.D. &Chapman, A.J.,"Analytical Solution of the Known-Heat-Load Space Radiator Problem", J. Spacecraft & Rockets, Vol.4, No.3, Mar. 1967.

15. Farvorskiy, O.N. & Kadaner, Y.S.,"Problems of Heat Transfer in Space', (Vysshaya Shkola Press, Moscow, 1972) NASA TT F-783, Aug. 1973.

16. Plamondon, J.A.,"Analysis of Movable Louvers for Temperature Control", J Spacecraft & Rockets, Vol. 1, No. 5, 1964.

17. Cotter, T.P.,"Theory of Heat Pipes", Los Alamos Scientific Lab., LA-3246-MS, Feb. 1965.

18. Marcus, B.D.,"Theory and Design of Variable Conductance Heat Pipes: Hydrodynamics and Heat Transfer", TRW13000-6021-RO-00, Res. Report No.1, April, 1971.

19. Stenger, F.J.,"Experimental Feasibility Study of Water- Filled CVapillary-Pumped Heat Transfer Loops", NASA TM X-1310, Nov. 1966.

20. Fried, E. & Costello, F.A.,"Interface Thermal Contact Resistance Problem in Space Vehicles", ARS Journal, Vol. 32, No. 2, Feb. 1962.

21. Chetty, P.R.K., "Satellite Technology and Its Applications," Tab Books Inc., Blue Ridge Summit, PA, 1988.

22. Wertz, J.R. edited,"Spacecraft Attitude Determination and Control", D. Reid Publishing Co., 1985.

Table 1 ELV commercial launch service costs

Scout	$ 10 - 15 M
Delta II	$ 40 - 50 M
Atlas I / IIA	$ 65 - 80 M
Titan III	$130 -140 M (Dedicated Flight)
Ariane II / III	$ 40 - 80 M
Ariane IV	$100 -110 M (Dedicated Flight)
Long March	$ 30 - 60 M
Proton	$ 30 - 65 M
Japanese H-II	$100 -120 M (Commercial Availability-TBD)

Table 2 Approximate weight carrying capacity (Kg) of ELV
- to a 100-NMI circular orbit -

Booster	Low Inclination		High Inclination	
	Wt (Kg)	(degree)	Wt (Kg)	(degree)
Scout	265	(2.9)	205	(90)
Titan II	2,180	(63.4)	1,680	(99)
Pegasus	440	(0)	350	(90)
Atlas I	5,800	(28.5)	-	
Atlas II	6,600	(28.5)	-	
Atlas IIA	7,000	(28.5)	-	
Titan III	14,500	(28.6)	-	
Delta 7920	5,000	(28.7)	3,800	(90)

Table 3 Elements in spacecraft cost model
of subststems and system level

- SPACECRAFT SUBSYSTEMS
  Electric power
  Thermal control
  Structure
  Auxiliary propulsion
  Attitude control
  Communication, command & data handling

- SPACECRAFT SYSTEM LEVEL	Protoflight	Follow-on Unit
Spacecraft integration & test	19%	34%
Program management	27	33
Systems engineering & integration	19	16
Product assurance	16	17
Ground support equipment	19	-

Table 4 Constants a & b in the cost models

Subsystem Name	ID	Protoflight				Follow-On			
		Non-Explorer		Explorer		Non-Explorer		Explorer	
		a	b	a	b	a	b	a	b
Electric Power	1	0.028	0.87	0.028	0.87	0.029	0.735	0.029	0.735
Thermal Control	2	0.104	0.619	0.104	0.619	0.104	0.706	0.029	0.706
Structure	3	0.186	0.405	0.165	0.653	0.01	0.757	0.01	0.757
Auxiliary Propulsion	4	0.089	0.8	0.089	0.8	0.087	0.559	0.087	0.559
Attitude Control	5	0.37	0.63	0.045	0.74	0.042	0.874	0.077	0.445
CC&DH	6	0.272	0.797	0.037	0.959	0.045	0.955	0.056	0.73
System Level	100	0.786	0.832	0.786	0.832	0.458	0.925	0.458	0.925

* The validity of the simplified expressions is subject to the weight ranges of individual subsystems shown in the weight limits table.
** The Structure subsystem was originally classified in two classes (1) Simple design, and (2) Complex design. The constants were entered in the "Non-Explorer" category for the "Simple design" and in the "Explorer" for the "Complex design."

Array Sizing and Weight Estimating

Fig. 1 Equations for solar array sizing and weight estimating.

Space Radiators Formulas

Fig. 2 Formulas for sizing a fin-tube space radiator.

Velocity Requirements and Propellant Weight Formulas

H_a Apogee altitude
H_p Perigee altitude
$\Delta H = H_a - H_p$ Altitude error

$V_{cp} = \sqrt{\dfrac{\mu}{Re + H_p}}$ Characteristic velocity of circular orbit at perigee

$V_p = \sqrt{2\mu\left(\dfrac{1}{Re+H_p} - \dfrac{1}{2Re+2H_p+\Delta H}\right)}$ Perigee velocity

$\Delta V_p = V_p - V_{cp}$ Req'd add'l velocity to establish the Hohmann ellipse

$V_a = V_p\sqrt{\dfrac{Re+H_p}{Re+H_p+\Delta H}}$ Required velocity increment to be imparted to satellite at apogee

$V_{ca} = \sqrt{\dfrac{\mu}{Re+H_p+\Delta H}}$ Characteristic velocity of circular orbit at apogee

$\Delta V_a = V_{ca} - V_a$ Req'd add'l velocity increment at apogee

$\Delta V_i = 2V_{ca}\sin\left(\dfrac{\theta}{2}\right)$ Velocity to correct the inclination error

$\Delta V_{T1} = \Delta V_p + \Delta V_a + \Delta V_i$ Req'd total velocity increment

$\Delta V_{al} = \sqrt{V_a^2 + V_i^2}$ Combined velocity increment for perigee raise & inclination correct.

$\Delta V_{T2} = \Delta V_p + \Delta V_{al}$ Alternative req'd total velocity

$W_{p1} = W_c\left[\exp\left(\dfrac{\Delta V_{T1}}{g \cdot I_{sp}}\right) - 1\right]$ Req'd propellant weight to correct the launch errors

$W_{p2} = W_c\left[\exp\left(\dfrac{\Delta V_{T2}}{g \cdot I_{sp}}\right) - 1\right]$ Alternative req'd propellant weight to correct the launch errors

Fig. 3 Formulas for determining required propellent for correction of attitude errors.

Protoflight S/C Cost Estimating Worksheet

Project: ABC   Date: 2/1/90

Type of Spacecraft: Non-Explorer   Structure Class: Simple

PROTOFLIGHT SPACECRAFT SUBSYSTEMS	a	W	b	Calculated Cost:	Known Cost:
ELECTRIC POWER	C= 0.028(	644.080) exp	0.870=	$7.779	$6.148
THERMAL CONTROL	C= 0.104(	121.180) exp	0.619=	$2.026	$2.359
STRUCTURE	C= 0.186(	531.930) exp	0.405=	$2.363	$2.566
AUXILIARY PROPULSION	C= 0.089(	77.770) exp	0.800=	$2.897	$2.897 *
ATTITUDE CONTROL	C= 0.370(	40.270) exp	0.630=	$3.796	$3.796 *
COMMU., COMMAND, AND DATA HANDLING	C= 0.272(	157.890) exp	0.797=	$15.369	$15.369 *
Total Subsystem Cost (Sum of Subsystem Costs)			K=	$34.231	$33.137

Note: Asterisk (*) indicates that the calculated cost was used for the known cost.

PROTOFLIGHT SYSTEM LEVEL

	a	W	b	Calculated	Known
Total System Level Costs	C= 0.786(	34.231) exp	0.832=	$14.861	$14.465
SPACECRAFT INTEGRATION AND TEST			@19%=	$2.823	$2.748
PROGRAM MANAGEMENT			@27%=	$4.012	$3.905
SYSTEMS ENGINEERING AND INTEGRATION			@19%=	$2.823	$2.748
PRODUCT ASSURANCE			@16%=	$2.377	$2.314
GROUND SUPPORT EQUIPMENT			@19%=	$2.823	$2.748
TOTAL PROTOFLIGHT COST (in 1980 million dollars)				$49.093	$47.60
TOTAL PROTOFLIGHT COST (in 1991 million dollars)				$92.491	$89.685

Fig. 4 Worksheet of cost estimate for a protoflight spacecraft.

Follow-On Unit S/C Cost Estimating Worksheet

Project: ABC   Date: 2/1/90

Type of Spacecraft: Non-Explorer   Structure Class: Simple

FOLLOW-ON UNIT SPACECRAFT SUBSYSTEMS	a	W	b	Calculated Cost:	Known Cost:
ELECTRIC POWER	C= 0.029(	644.080) exp	0.735=	$3.364	$6.148
THERMAL CONTROL	C= 0.026(	121.180) exp	0.706=	$0.828	$2.359
STRUCTURE	C= 0.010(	531.930) exp	0.757=	$1.157	$2.566
AUXILIARY PROPULSION	C= 0.087(	77.770) exp	0.559=	$0.991	$0.991 *
ATTITUDE CONTROL	C= 0.042(	40.270) exp	0.874=	$1.061	$1.061 *
COMMU., COMMAND, AND DATA HANDLING	C= 0.045(	157.890) exp	0.955=	$5.657	$5.657 *
Total Subsystem Cost (Sum of Subsystem Costs)			K =	$13.061	$18.786

Note: Asterisk (*) indicates that the calculated cost was used for the known cost.

FOLLOW-ON UNIT SYSTEM LEVEL

	a	W	b	Calculated	Known
Total System Level Costs	C= 0.458(	13.061) exp	0.929=	$4.984	$6.986
SPACECRAFT INTEGRATION AND TEST			@34%=	$1.694	$2.375
PROGRAM MANAGEMENT			@33%=	$1.644	$2.305
SYSTEMS ENGINEERING AND INTEGRATION			@16%=	$0.797	$1.117
PRODUCT ASSURANCE			@17%=	$0.847	$1.187
TOTAL FOLLOW-ON UNIT COST (in 1980 million dollars)				$18.046	$25.77
TOTAL FOLLOW-ON UNIT COST (in 1991 million dollars)				$33.999	$48.555
TOTAL PROJECT COST (Number of Follow-On Units 3)				$194.48	$235.35

Total cost does not include prime contractors fee.

Total cost = Number of follow-on units X follow-on unit cost + protoflight cost.

Fig. 5 Worksheet of cost estimate for follow-on units.

File	Edit	Environment	Design	Structure
New Database...	Show Clipboard	Design	Structure	New File...
Open Database...		User	Layout...	
Save Structure		Custom	Procedure...	
Preferences...			Menu...	
Page Setup...			Password	
Print...				
Quit				

Fig. 6 The main menubar after selecting the DESIGN environment.

File	Edit	Environment	Enter	Select
New Database...	Select All	Design	New Record	Show All
Open Database...	Show Clipboard	User	Apply Formula...	Search...
Import Data...		Custom		Search and Modify...
Export Data...				Search by Formula...
Choose File/Layout...				Sort...
Page Setup...				
Print...				
Quit				

Report	Special
Quick...	Edit ASCII Map...
Labels...	Execute Procedure...
Graph...	

Fig. 7 The main menubar after selecting the USER environment.

File	Edit	System Info	Subsystem Data	Subsystems
Quit	Show Clipboard	Input System Info	Input New Data	Electric Power
		Modify System Info	Modify Data	Thermal Control
		Delete System Info	View Data	Structure
			Delete Data	Auxiliary Propulsion
				Attitude Control
				CC & DH

Formats	Tables
Summary of S/C System	Cost Constants
Cost Models	Print Cost Constants
System Totals	Inflation Factors
	Print Inflation Factors
	Physical Constants
	Print Physical Constants
	Weight Limits
	Print Weight Limits
	Modify Table

Fig. 8 The main menubar after selecting the CUSTOM environment.

File	Edit	Electric Power	Tables	Formulas
Quit	Show Clipboard	Input Data View Data Modify Data Delete Data	Solar cell Table	Sunlight and Eclipse Durations Array Sizing and Weight Temperatures Battery Size and Weight

File   Edit   Thermal   Formulas   Tables

File	Edit	Thermal	Formulas	Tables
Quit	Show Clipboard	Input Data View Data Modify Data Delete Data	Simple Space Radiator Sizing Space Radiator Sizing Thermal Louvers and Heaters	Conductive Paints Table Thermal Coatings Table Black Coatings Table White Coatings Table Miscellaneous Coatings

File   Edit   Structure   Formulas   Tables

File	Edit	Structure	Formulas	Tables
Quit	Show Clipboard	Input Data Modify Data View Data Delete Data	Structure Variables*	Structural Material Table

File   Edit   Auxiliary Propulsion   Formulas   Tables

File	Edit	Aux.Propulsion	Formulas	Tables
Quit	Show Clipboard	Input Data Modify Data View Data Delete Data	Vel. & Weight Requirements	Message*

File   Edit   Attitude Control   Formulas   Tables

File	Edit	Attitude Control	Formulas	Tables
Quit	Show Clipboard	Input Data Modify Data View Data Delete Data	Message*	Message*

File   Edit   CC&DH   Formulas   Tables

File	Edit	CC&DH	Formulas	Tables
Quit	Show Clipboard	Input Data Modify Data View Data Delete Data	Message*	Message*

Fig. 9 The menubars when individual subsystems are selected.

Fig. 10 The AEOSS output for sizing a solar array.

Fig. 11 The AEOSS output for sizing a fin-tube space radiator.

Fig. 12 The AEOSS output for determining the required propellant.

INVESTIGATION OF VISUAL INTERFACE ISSUES IN SPACE TELEOPERATION USING A VIRTUAL TELEOPERATOR

M. A. Machlis*
H. L. Alexander**
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

Abstract

A simulator has been developed to examine human factors issues in teleoperation. A graphic workstation simulates the visual feedback which would be provided to an operator by vehicle-mounted video cameras on an actual teleoperator. The software design allows easy modification of vehicle dynamics and content of the simulated environment. Command input is via a combination of hand- and foot-controllers, and visual feedback is provided by a CRT monitor or a VPL Eyephones stereoscopic head-mounted display. A mechanical-linkage head tracker allows transformation of views based on operator head orientation. Using the head-mounted display with head-slaved views was found to provide a strong sense of telepresence. A representation of the body of the teleoperator was added to the visual scenes. It was expected that, when using the head-mounted display with head-slaved views, this would reduce operator disorientation by providing precise visual cues to gaze direction. Preliminary results indicate that including a vehicle body does reduce disorientation and increases performance on some tasks.

Introduction

Telerobotics and teleoperation have become areas of intense research in recent years. These technologies allow remotely-operated vehicles to replace humans in performing specific tasks where the environment is in some way hostile to human life. Such environments include high-radiation areas of nuclear power plants, certain situations dealing with armed criminals, surfaces of distant planets, and Earth orbit. Teleoperators, under the direct control of human operators, are used when the use of autonomous telerobots is not feasible due to task complexity or uncertainty in task structure.

Teleoperated vehicles built to operate in the different environments mentioned above differ greatly in general design and capabilities. They do, however, all have some kind of operator-interface station where the operator inputs commands to the vehicle and receives feedback on its state. In many cases the primary feedback path is visual -- the operator is presented with an image or images from one or more video cameras mounted on the teleoperator.

We have developed a teleoperation simulator which receives commands from an operator, calculates the motion of the simulated vehicle, and generates graphic images representing feedback from a single camera or stereo pair of cameras mounted on the vehicle. The images are generated based on a predefined virtual environment in which the simulated vehicle maneuvers. The simulator is designed to allow easy modification of the control input devices, visual display type, vehicle dynamics, and virtual environment.

This paper describes an ongoing research project using the teleoperation simulator to evaluate several different visual display configurations. Preliminary experiments simulating one-, three-, and six-degree-of-freedom (DOF) vehicles operating in two- and three-dimensional space have been completed. Display modes examined so far have been: CRT monitor (monoscopic image), head-mounted display (stereoscopic images) with fixed views, and head-mounted display (stereoscopic) with head-slaved views. An additional experimental variable was whether or not a representation of the teleoperator body was added to the visual scene. Statistical analysis of performance figures in future experiments should reveal the relative influence of each variable on performance for each task.

Background

In the field of teleoperation, many research groups are currently focussing their attention on the concept of telepresence. The goal of telepresence is to deliver sensory information to and receive command inputs from the human operator in as "natural" a manner as possible. Sensory feedback and command formats are designed to mimic sensory input/physical motions that humans experience in everyday life. A basic tenet of telepresence is that the stronger the operator's sense of "being there" -- the feeling that he or she is actually being subjected to the motions of the teleoperator rather than observing and controlling them from a distance -- the better the operator's performance will be. With a strong sense of telepresence the operator is more easily able to draw on his or her natural sensory processing and motor skills. Although sensory information inherent to the concept of telepresence includes visual, aural, tactile, and proprioceptive, the discussion here is restricted to visual information.

---

*Research Assistant, Department of Aeronautics and Astronautics

**Bradley Career Development Professor, Department of Aeronautics and Astronautics. Member AIAA

The earliest teleoperators were telemanipulators developed to handle radioactive material from a distance. These were operated in a direct view configuration: the operator was located near the manipulator, controlling it and observing its motions firsthand. The first non-direct view teleoperators, limited by available technology, used video cameras and fixed television displays to provide visual feedback to the operator.[1] As practical head-mounted displays became available, it was a natural extension of telepresence to use these displays in teleoperation, as they permit presentation of wide field of view, stereoscopic images. With the addition of head-tracking systems, which calculate the orientation of the operator's head in real time, the perspective of the presented images can be slaved to the operator's head motion. This permits the operator to look around within the environment of the teleoperator without having to rotate the entire vehicle.

There has been some debate over the effects of stereoscopic views on visual perception in remote vision. Results reported by Tharp et al. state, however, that in teleoperation stereoscopic views provide a better sense of telepresence than monoscopic views,[2] which should increase performance on most tasks. Evidence obtained by Pepper and Hightower support this fact by demonstrating better performance with a stereoscopic as opposed to monoscopic presentation on two specific tasks.[3] Additional research by Pepper et al. indicates that, when using a head-mounted display, stereoacuity is improved under both stereoscopic and monoscopic viewing conditions when the images are isomorphically transformed by the operator's head movements.[4] All of this evidence supports the intuitive idea that a head-mounted display with head-slaved views gives the strongest possible visual sense of telepresence, and should therefore result in superior task performance in teleoperation as compared to other visual display modes.

The results cited above all involve telemanipulation tasks with a fixed frame of reference. Another important aspect of teleoperation is navigation of a teleoperator from one point to another. This will involve some combination of target acquisition and identification, obstacle avoidance, and precise vehicle control. Although a head-mounted display with head tracking gives superior performance in telemanipulation tasks, it is not yet clear how it will influence performance on navigation tasks. As described by Pepper et al., "... helmet mounted stereo TV display might also place additional demands on the operator which may or may not be offset by performance gains associated with the added degree of complexity and sophistication" (p. 173).[4]

Hypothesis

The study described in this paper is concerned with operator disorientation in teleoperator navigation when using a head-mounted display with head-slaved views. In this display configuration the transformation between the operator's command inputs to the vehicle and the visual scene presented on the display is a combination of two elements: the motion of the vehicle itself and the orientation of the operator's head. The operator has indirect access to both the combined transformation, through the interpretation of the visual scene, and the head-orientation transformation, through proprioceptive and vestibular sensors. In order to control the vehicle precisely, however, the operator needs an accurate estimate of the motion of the vehicle. The hypothesis of this paper is that in tasks which require significant head movement, the operator often will not be able to accurately determine the motion of the vehicle from the available information, and therefore will not be able to control the vehicle as precisely as when using a display with fixed views. This may, depending on the nature of the task, negate the performance increase gained with head-slaved views from the ability to visually scan the environment by rotating one's head.

It is proposed that adding a representation of the vehicle body to the visual field will provide cues which will allow the operator to more easily and accurately estimate the motion of the vehicle. The vehicle body is expected to provide more explicit information on head orientation, as well as a fixed reference -- when the head is held motionless -- which will allow more accurate estimates of the velocity of the visual field.

The disorientation problem under discussion is similar to one which has been described by pilots of high-performance fighter aircraft. The raised bubble canopies of these aircraft make a very large field of view of the outside environment visible to the pilot. When gazing out through the canopy, the pilot often will not have any part of his own aircraft in his or her view other than the clear canopy, which provides no orientation reference. Pilots have reported becoming confused -- specifically, losing their sense of the orientation of their aircraft with respect to themselves and to the outside world -- while scanning or tracking targets across this large field of view. To counter this, some have attached strips of tape to the inside of their canopies which provide an orientation reference when no other part of the aircraft is in sight.[5]

Teleoperators designed for navigation through environments which utilize head-slaved camera pointing often have their camera(s) placed such that no part of the vehicle blocks any potential field of view of the camera(s). This allows maximum viewing of the environment, but may also induce the sense of disorientation described above. When directing his or her gaze away from the "natural" line of sight (directly forward of the vehicle), the operator may lose track of the orientation of the teleoperator with respect to the view he or she is seeing and with respect to the vehicle's environment.

The superimposition of a computer-generated vehicle body onto the video feedback from a teleoperator may suppress the disorientation tendency described above. The

viewing transformations needed to generate this body image in real time depend only on the operator's head orientation -- data which is readily available. This method is preferable to placing parts of the vehicle in the camera's field of view because the computer-generated body may be a wire-frame or translucent model which does not block the operator's view of the environment.

System Description

The primary component of the teleoperation simulator is a Silicon Graphics IRIS 4D/25 graphics workstation, which computes the simulated vehicle's dynamics and generates the graphic images. The IRIS's hardware is optimized for graphics applications, and it has a library of graphics subroutines for drawing both perspective and orthographic three-dimensional views. The IRIS is connected via a serial communication line to a 20 MHz IBM PC-compatible 80386-based microcomputer. The control devices and head tracker are connected to an input/output (I/O) box, which contains a power supply and an interface to an analog-to-digital signal conversion board in the PC. The PC reads voltages from the I/O box, interprets them, and sends command input and head orientation values to the IRIS. Fig. 1 shows the organization of the system.
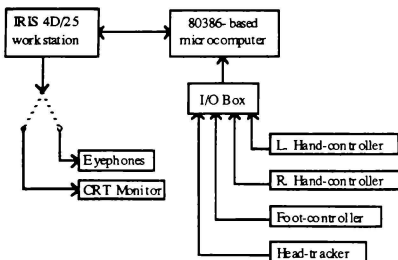
Fig. 1. Block diagram of teleoperator simulation system

The current configuration of control input devices allows the operator to use any combination of two three-DOF displacement-sensing hand-controllers and one two-DOF force-sensing foot-controller for simultaneous control of up to eight DOF. Experiments have been carried out comparing performance using the two hand-controllers versus a combination of hand-controllers and foot-controller in controlling a six DOF vehicle.[6]

The video signal from the IRIS can be displayed on any of three video devices. The first option is the large high-resolution Silicon Graphics monitor sold with the IRIS. The second is a standard NTSC-format RGB monitor. Both monitors can display color images. The

third display alternative is a VPL Eyephones head-mounted display system. The Eyephones consist of a pair of color liquid-crystal display screens with focussing optics mounted in a unit which is worn on the head. The screens have a resolution of 360 x 240 pixels, and the optics create a field of view of 80 degrees horizontal by 60 degrees vertical for each eye.[7] Each screen is seen by one eye, with some overlap in the field of view of the two eyes to allow for stereoscopic viewing. When using the Eyephones the NTSC-format RGB video signals from the IRIS are split: the red signal is connected to the left eye screen and the blue signal is connected to the right eye screen. This allows the IRIS to generate images for both eyes simultaneously, with the drawback that only monochrome images can be displayed. Fig. 2 shows an IRIS-generated rectangular tunnel as seen through the Eyephones.
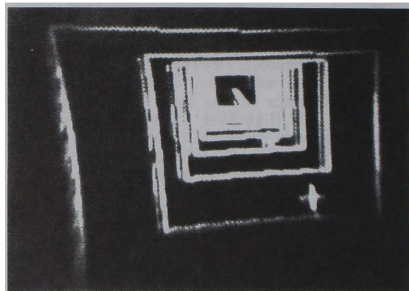
Fig. 2a. Left eye image of computer-generated tunnel as seen through Eyephones
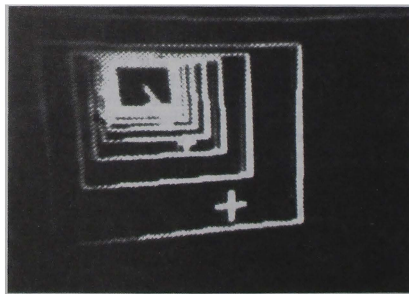
Fig. 2b. Right eye image of computer-generated tunnel as seen through Eyephones

Head tracking is accomplished by a mechanical-linkage system designed and built in-house (Figs. 3,4). The shaft of a high-linearity potentiometer serves as the rotation axis of each joint. A fixed voltage difference is applied to the end taps of each potentiometer, and the voltage at the center tap is linearly related to the rotation angle of the shaft, and therefore of the joint itself. One end of the head-tracker linkage is attached to a fixed base, while the opposite end is attached to the head-mounted display. The head-tracker has a total of six joints to allow computation of the full three-dimensional position and orientation of the subject's head with respect to the base. A mechanical system was chosen over commercially available magnetic systems because of the time delay inherent in the magnetic systems. If it is desired to simulate a system with delay -- simulating, for example, the communications delay in ground-based teleoperation of a vehicle in Earth orbit -- the delay can easily be implemented in software. The mechanical system also has higher angular resolution and precision than the Polhemus magnetic system which is shipped with the Eyephones.

Fig. 4. Subject operating hand-controllers, wearing Eyephones with head-tracker
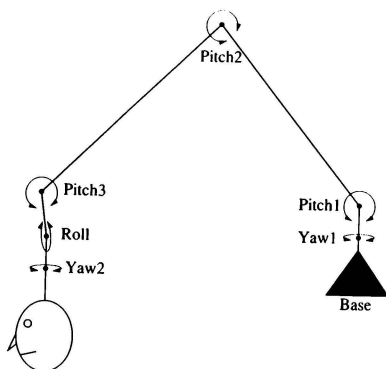
Fig. 3. Head-tracker joint arrangement

The simulator can reproduce a wide variety of environments and vehicle dynamics. The main constraint is that, because of speed limitations of the IRIS, all objects are drawn as wire-frame models. This eliminates the visual cues related to solid surfaces and shading. The wire-frame environments are nevertheless sufficiently visually rich to provide effective motion cues. This is supported by the fact that several subjects have reported experiencing moderate motion sickness after operating the simulator using the head-mounted display.

Experiments

Experiments on the teleoperation simulator are currently being designed to examine the effects of a superimposed vehicle body on teleoperator navigation tasks. The conditions to be examined include: monoscopic monitor display, stereoscopic head-mounted display with fixed view, and stereoscopic head-mounted display with head-slaved view. Performance data will be taken on each condition with and without a superimposed vehicle body.

In order to keep the resolution of the different display cases similar, and because of the relatively low resolution of the Eyephone screens, the NTSC RGB monitor rather than the Silicon Graphics monitor will be used in the monoscopic monitor display condition. In the head-slaved view case, images will be transformed by the rotational orientation of the operator's head; translation of the head will not not reproduced. This will simulate a common head-slaved camera motion system on modern teleoperators.

Although no statistical data has yet been collected, preliminary experiments have been performed with various task environments and different vehicle dynamics which have provided much qualitative performance data and anecdotal information.

One environment used was an extension of that used by Cinniger[6]. Each of the six translational and rotational accelerations of the vehicle was set directly proportional to the deflection of a particular axis on one of the hand-controllers, simulating a thruster-propelled vehicle in three-dimensional space with no external forces. The environment consisted of a series of rectangles in space modified to provide additional orientation cues (Fig. 5). Stars were added in the distance to provide cues to rotational motion. The subjects' task was to fly along the trajectory defined by connecting the centers of the

rectangles -- crossing the plane of each rectangle as close to its center as possible -- as quickly as possible.
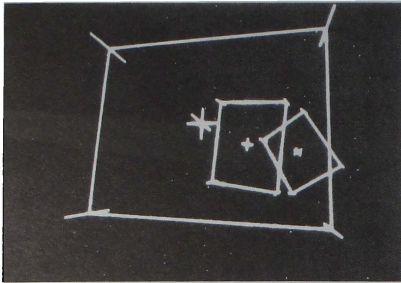
Fig. 5. Fly-through rectangle environment

Fig. 6a. "Aerial" view of line-of-columns environment

Fig. 6b. Operator's view of line-of-columns environment, with vehicle body

While this experiment did provide some useful data, there was a wide performance variation among subjects. Some were able to complete the task fairly quickly and accurately, while others were only marginally able to complete the task at all. The data obtained were highly distributed and not useful in determining the influences of the different experimental variables on performance.

A simpler configuration, designed to obtain more focussed data, provided only one controllable DOF to the subject. This task involved driving a vehicle on the surface of a planar grid. A row of unevenly-distributed columns was laid out across the grid (Fig. 6). The dynamics were made to simulate driving an automobile-type vehicle at constant velocity: forward speed was fixed and the turning radius was made inversely proportional to the deflection of one hand-controller axis. The hand-controller thus acted as a steering wheel for the vehicle. The vehicle body representation in this configuration and both configurations below resembled an automobile, with a framed windshield and a hood in front of the operator's eye location and window outlines to the sides of and behind the eye point.

In this environment the subjects' task was to slalom through the line of columns: driving to the right side of the first column, crossing over to pass to the left of the second, back to the right side of the third, and so on. An examination of trajectory plots for different subjects for the different visual displays revealed that for most subjects, after a small amount of practice, there was very little variation between cases. The task was too easy and could be performed very well by most subjects under all test conditions.

Experiments with three controllable DOF were also performed with the line-of-columns environment and slalom task. Vehicle motion was governed by two-dimensional inertial dynamics with no external forces present. The operator controlled translational accelerations along the fore/aft and left/right axes and rotational acceleration around the vertical (yaw) axis. While one subject could not control the vehicle well enough to complete the task, the others performed the task very well under all display configurations, again resulting in little useful data. The subjects who performed well kept their velocity along the line of columns and their yaw angle constant, while using only lateral acceleration to complete the task, so that they only actually controlled one DOF.

A setup similar to that above forced the subjects to control all three DOF by rearranging the columns into a square with three columns per side. Fig. 7 shows a subject's view of this environment with and without the

superimposed vehicle body. The subjects were instructed to slalom around the columns, steering outside the corner columns and inside the side columns. Although quantitative performance data have not yet obtained, preliminary examination of trajectory data tends to support the hypothesis. Fig. 8 plots the trajectories followed by a subject using the head-mounted display with head-slaved views, with and without the vehicle body. The diamonds in the figure indicate the locations of the columns. The figure shows that when operating with the superimposed vehicle body, the subject flew more smoothly and consistently. When navigating without the vehicle body the paths are more jagged, due to the fact that the operator made many more corrections to his trajectory. The operator also reported becoming disoriented more often and took longer to complete the task in the latter case.

Fig. 8a. Trajectory plot, run #1, with vehicle body
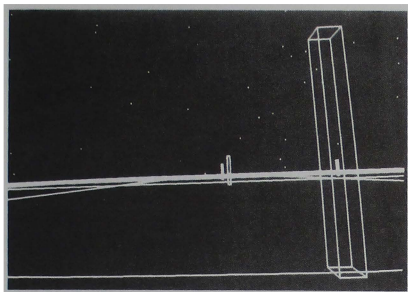
Fig. 7a. Operator's view of square-of-columns environment, without vehicle body
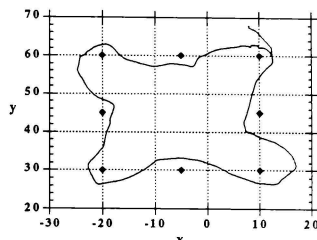
Fig. 8b. Trajectory plot, run #2, with vehicle body

Fig. 7b. Operator's view of square-of-columns environment, with vehicle body
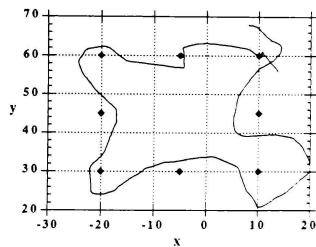
Fig. 8c. Trajectory plot, run #1, without vehicle body
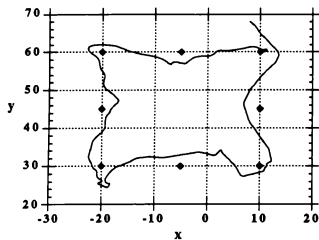
Fig. 8d. Trajectory plot, run #2, without vehicle body

Conclusions

Although much of the evidence gathered so far has been qualitative or subjective in nature, it supports the original hypothesis. It seems likely that for many teleoperator navigation tasks, the best visual display system is also the most "natural" -- head-slaved images presented via a head-mounted display, with an overlaid vehicle body. Future experimentation with the teleoperation simulator will involve collecting quantitative, statistical evidence to prove this more conclusively. Future work will also include experimenting with alternate display methods: stereoscopic monitor displays using colored filters and a monitor-as-window-to-the-environment configuration making use of the head tracker.

References

1. Coiffet, P., gen. ed. Robot Technology. 8 vols. Englewood Cliffs, N.J.: Prentice-Hall, Inc., 1986. Vol. 3A: Teleoperation and Robotics: Evolution and Development, by J. Vertut and P. Coiffet.

2. Tharp, G.; Liu, A.; Yamashita, H.; Stark, L.; Wong, B.; and Dee, J. "A Helmet-Mounted Display to Adapt the Telerobotic Environment to Human Vision." Proceedings of the Third Annual Workshop on Space Operations Automation and Robotics (1989): 477-481.

3. Pepper, R. L. and Hightower, J. D. "Research Issues in Teleoperator Systems." Proceedings of the Human Factors Society - 28th Annual Meeting (1984): 803-807.

4. Pepper, R. L.; Cole, R. E.; Spain, E. H.; and Sigurdson, J. E. "Research Issues Involved in Applying Stereoscopic Television to Remotely Operated Vehicles." Proceedings of SPIE - The International Society for Optical Engineering 402 (1983): 170-173.

5. Gillingham, Kent. Brooks School of Aerospace Medicine. Personal discussion.

6. Cinniger, A. G. "Control Interfaces and Handling Qualities for Space Telerobotic Vehicles." Master's Thesis, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, 1991.

7. VPL Research, Inc. "Eyephone Operation Manual." 1989.

USE OF PILOTED SIMULATION FOR HIGH-ANGLE-OF-ATTACK AGILITY RESEARCH AND DESIGN CRITERIA DEVELOPMENT

Marilyn E. Ogburn* and John V. Foster
NASA Langley Research Center
Hampton, VA

Keith D. Hoffler**
ViGYAN, Inc.
Hampton, VA

ABSTRACT

This paper reviews the use of piloted simulation at Langley Research Center as part of the NASA High-Angle-of-Attack Technology Program (HATP) program to provide methods and concepts for the design of advanced fighter aircraft. A major focus of this program is to develop the design process required to fully exploit the benefits from advanced control concepts for high-angle-of-attack agility. Important methodologies associated with the effective use of piloted simulation for this research are described, particularly those relating to the test techniques, validation of the test results, and design guideline/criteria development.

INTRODUCTION

Projected scenarios for future air combat indicate the need for highly agile aircraft that can operate effectively over a greatly expanded maneuvering envelope. In response to this need, significant activities are currently underway to develop technologies that are key to providing this enhanced capability. These technology areas include high-angle-of-attack aerodynamics, high-angle-of-attack controls, propulsion systems, pilot/vehicle interface, and weapons. The National Aeronautics and Space Administration (NASA) is actively engaged in these efforts, with a major goal of developing flight dynamics technology to provide enhanced agility and handling qualities at high angles of attack that will enable aircraft to perform maneuvers that can be very advantageous in air combat (see figure 1). This capability can be achieved through the use of advanced control concepts such as vectoring of the engine thrust and unconventional aerodynamic devices that provide significant improvements in effectiveness, especially at high angles of attack.

A key NASA program which was conceived to address these advanced technology opportunities for high-performance aircraft is the High-Angle-of-Attack Technology Program (HATP). The HATP is a fighter technology development and validation program which is focusing on providing flight-validated methods and concepts essential for the design of fighters possessing unprecedented high-angle-of-attack maneuverability and controllability. The program uses the unique expertise and facilities of NASA's aeronautics research centers, including the Langley, Ames, and Lewis Centers. The research approach being taken is a balanced one involving closely-integrated wind-tunnel experiments, computational aerodynamics, piloted simulation, and flight tests of an F-18 research testbed airplane known as the High-Angle-of-Attack Research Vehicle (HARV). This vehicle has been modified to make it capable of

testing advanced controls, including multi-axis thrust-vectoring and advanced aerodynamic controls. Reference 1 contains a more complete description of this program

Piloted simulation has been an integral and key element of high-performance aircraft flight dynamics research at NASA Langley. Research activities have addressed flying qualities, control system design and effects, design guidelines development, and pilot/vehicle interface. The primary objectives of these simulator studies are to: (1) define and quantify the enhancements in agility provided by advanced control concepts under realistic combat conditions, (2) develop agility/handling qualities design requirements, including tradeoffs, for control laws, control effectiveness, and cockpit information systems, and (3) develop the design tools and methodology to enable these requirements to be met, so that the enhanced high-angle-of-attack capabilities can be effectively exploited. More that 20 years of experience with the application of piloted simulation to high-angle-of-attack flight dynamics have shown it to be a very effective research approach. As a result, piloted simulation is playing a major role in the high-angle-of-attack technology development process in the NASA High-Angle-of-Attack Technology Program. The primary facility used for this piloted simulation research is the Langley Differential Maneuvering Simulator (DMS), a fixed-based simulator which has the capability of simultaneously simulating two airplanes as they maneuver with respect to one another. The capability to simulate one-versus-two air combat is also provided by the use of a smaller dome facility known as the General Purpose Fighter Simulator (GPFS) in conjunction with the two DMS domes, as shown in figure 2. This piloted simulation agility research is illustrated in figure 3.

This paper presents an overview of the use of piloted simulation at NASA Langley for the development of high-angle-of-attack technologies as part of the NASA HATP program. The following sections describe the simulation methodologies used in the conduct of the tests, the validation of the test results, and specific methods used for design guideline/criteria development. Example results from recent research using piloted simulation are presented

when appropriate to illustrate the use of these methodologies. Some of these examples are drawn from agility research which was conducted to investigate the use of a preliminary thrust-vectoring concept for the F-18 HARV. Other examples are from a generic program in which candidate design guidelines for nose-down pitch control margin for relaxed static stability combat aircraft were developed. This pitch control margin research is described in reference 2.

NOMENCLATURE

b	wing span, ft
C_l	rolling moment coefficient
C_m	static pitching moment coefficient
C_{m_q}	pitch damping coefficient
C_m^*	minimum nose-down pitching moment coefficient at any α
C_n	yawing moment coefficient
h	altitude, ft
M	Mach number
p,q,r	body-axis roll, pitch, and yaw rates, deg/sec
$\hat{p},\hat{r}$	non-dimensional body-axis roll and yaw rates, $\dfrac{pb}{2V}$ or $\dfrac{rb}{2V}$
p_w	wind-axis roll rate, deg/sec
$\dot{q}$	pitch acceleration, rad/sec^2
S1, S2	slope of C_m versus α curve for α below and above $\Delta\alpha^*$, respectively, per deg
t	time, sec
$t\,|\,\Delta\phi|$	time to roll through a bank angle change, sec
T/W	thrust-to-weight ratio
V	free-stream velocity, ft/sec
X,Y	airplane body axes
α	angle of attack, deg
α^*	maximum α at which C_m^* occurs, deg
$\Delta\alpha^*$	range of angle of attack over which C_m^* occurs, deg
β	angle of sideslip, deg

γ	velocity vector pitch angle from horizontal, deg
δ_a	differential aileron deflection, positive for left roll, deg
δ_r	rudder deflection, positive for left yaw, deg
ε	tracking error, deg
θ, ϕ	pitch and roll angles, deg
$\Delta\phi_w$	change in wind-axis roll angle, deg
Ω	angular rotation rate, deg/sec
Subscripts:	
max	maximum value
o	initial value
rec	value for recovery to low angles of attack

TEST TECHNIQUES

Overall Research Process

The overall approach used to conduct piloted simulation studies is illustrated in figure 4. The application of piloted simulation to flight dynamics studies is, of course, dependent on the development of a valid mathematical model which generates accurate flight motions and handling qualities. Data obtained in static and dynamic wind-tunnel tests are used to develop aerodynamic math models for the studies. Although the model tests provide much information on high-angle-of-attack characteristics, they do not allow for a quantitative pilot evaluation of the flying qualities of the full-scale airplane during representative air combat maneuvering. Using the math model data, analysis can be performed prior to the piloted evaluation to characterize the aircraft stability characteristics and maneuvering capabilities as an aid in the interpretation of the results. The simulation validation process involves the use of ground-based testing and correlation with full-scale flight tests. Once the simulation fidelity has been established the piloted evaluation can proceed with added confidence. If appropriate flight test results are available they can be used as an aid in the evaluation process to determine the suitability of the evaluation maneuvers and other aspects of the evaluation methodology. As preparation for flight tests, piloted simulation is extremely useful for developing appropriate maneuvers and providing pilots the

opportunity to practice the required maneuver techniques prior to flight. The following sections will describe many of these simulation techniques and methodologies.

Desired Simulator Capabilities

The use of piloted simulation at Langley for high-angle-of-attack studies evolved from the initial use of a simple, single cockpit with a limited visual display (the GPFS) to the present twin-dome DMS. Early simulation efforts with the simple hardware identified several important simulator characteristics. Results of these studies indicated that in order to obtain a realistic evaluation of high-angle-of-attack flight characteristics, the simulation must present the pilot with a realistic air combat maneuvering environment. By providing a wide-angle visual display, air combat engagements could be simulated which required the pilot to be almost constantly looking outside of the cockpit to acquire and maneuver against an adversary; therefore, his opinion of the flying qualities and maneuvering capability would be based on similar visual information as in flight. In addition, it was found that there must be provided a good simulation of the cockpit environment in terms of pilot visibility, the display of flight instruments, and the use of a realistic force-feel system for the pilot stick and rudder pedals. Reference 3 describes some of these early piloted simulation studies.

As the simulation work at Langley progressed, the DMS was employed to meet these required characteristics. The DMS is a twin-dome fixed-base simulator with many state-of-the-art features which enhance its utility as a research tool. It has a number of capabilities which provide a realistic maneuvering environment for the pilot and allow for flexibility and repeatability of maneuvering conditions, and it has other capabilities which are necessary for high-angle-of-attack research. A computer-generated imaging (CGI) system provides a high-definition wide-angle visual scene with rotational and translational cues for the pilot. CRT displays and a head-up display (HUD) provide information within the cockpit (see figure 5). One-versus-one air combat engagements can be simulated by using both DMS domes, and one-versus-two capability is also available by using the third,

smaller GPFS dome. As many as two target images can be provided for each of the three domes using laser-generated or airplane model images with the proper apparent size, location, and orientation. The cockpits are equipped with a conventional center stick, rudder pedals, and a throttle. Provisions can be made for other pilot controls if required. A hydraulic force-feel system provides desired stick and pedal force and dynamic characteristics. Reference 4 contains a detailed description of the DMS.

Software Requirements

Aerodynamic Math Model. - In the development of a valid mathematical model for high-angle-of-attack simulation studies of specific configurations, sufficiently accurate models of the engine and flight control system are relatively easy to define. The ability to accurately predict high-angle-of-attack motions, however, is also highly dependent on the accuracy of the math model used to represent the aerodynamics during complex maneuvering. The aerodynamic modeling is the most difficult aspect of the high-angle-of-attack math model development, due to the extremely complex nature and configuration dependence of the flow phenomena at these conditions. Comprehensive, non-linear data bases are required to accurately represent these high-angle-of-attack aerodynamic characteristics. A major concern is that the mathematical modeling for the prediction of these motions is highly dependent on the results of wind-tunnel tests for the required static and dynamic aerodynamic data. The aerodynamic modeling accuracy will therefore only be as good as the accuracy of the wind-tunnel results and the accuracy of the application of these results to the math model.

In the past, conventional math models incorporating extensive data bases which combine static and small-amplitude damping wind-tunnel results have been applied with some success due to the fact that the simulated aircraft were quite limited in their ability to maneuver at stall/post-stall angles of attack because of poor control effectiveness (refs. 5 and 6). It is projected that technologies currently being explored will enable future fighters to have a greatly expanded high-angle-of-attack maneuvering envelope. Furthermore, these

aircraft will have the capability of generating rapid angular motions throughout this enlarged envelope. Figure 6 conceptually illustrates the anticipated increases in maximum pitch- and roll-rate capability in an expanded angle-of-attack envelope. The ability to accurately predict these motions using mathematical models presents a most difficult challenge for the flight dynamicist. For these highly agile combat aircraft, recent results have shown that conventional aerodynamic math models may be deficient in correctly representing these aerodynamics. In particular, certain phenomena such as wing rock are not yet understood well enough to be modeled with high accuracy. The impact of incorporating additional terms in the modeling of high-angle-of-attack aerodynamics is being investigated. Examples of these terms include those which account for dynamic stall phenomena during pitch maneuvers and those which represent the effects due to steady rotational motions about the velocity

vector (rotary derivatives) and lateral accelerations ($\dot{\beta}$ derivatives) during rolling conditions.

Large-amplitude aircraft maneuvers, however complex, can essentially be broken down into combinations of simple characteristic maneuvers. As is illustrated in figure 7, three basic characteristic maneuvers are: (1) pure pitch motion about the aircraft Y axis, (2) a constant angle-of-attack roll about the velocity vector, and (3) a pure sideslip motion. The first two types of maneuvers are the focus of current modeling studies at Langley. Reference 7 describes these investigations. The ability to roll effectively at high angles of attack is of particular importance to combat aircraft. There is concern that conventional math models which represent the dynamic effects by linear derivatives may not adequately represent the aerodynamics associated with rapid, large-amplitude coning rolls at high angles of attack that future highly agile aircraft will be able to perform. As a first step in investigating potential refinements to the aerodynamic math models, incorporation of rotary balance wind-tunnel data was studied. Assessment of the potential effects of this model refinement was made by comparing calculated motions from a six-degree-of-freedom simulation using both types of aerodynamic models. The simulation was of a representative current

fighter airplane for which static, forced-oscillation, and rotary balance wind-tunnel data had been obtained. Figure 8 compares the time history responses to a maximum pilot roll command at $\alpha_0 = 35°$ using the conventional model and the rotational model. The results show substantial differences in the time histories of aircraft maneuver states such as sideslip and angular rates as well as control deflections. These results suggest that refinements to the currently used conventional aerodynamic models may be necessary to more accurately predict the maneuver performance, stability, and controllability of future highly agile aircraft.

Flexibility. - Another key software requirement for piloted simulation studies is flexibility in the model for the purpose of examining the effects of parametric variations of various aircraft characteristics. Past simulation studies at Langley have involved the variation of performance, flight control law and control system characteristics, stability characteristics, and control effectiveness. Often, these variations can be easily implemented by assigning to a variable name a numerical value which can be changed at will. However, in some cases, multipliers or extrapolations which are functions of some variable or a completely different representation may be required. The purposes of these investigations have been to: (1) assess the effect of airframe and engine modifications and advanced control concepts on the stability characteristics and/or maneuvering performance, (2) develop flight control laws to effectively utilize high-angle-of-attack maneuvering capability, and (3) develop design criteria for control laws and control effectors. An example of a significant agility result obtained from a simple parametric variation is shown in figure 9. The sea-level static thrust-to-weight ratio of a configuration with thrust-vectoring (TV) controls was varied to evaluate the effect of thrust changes on the enhancements in maneuvering capability due to their use in rapid nose-up pitch maneuvers. A maximum pitch command was applied from 1g trim conditions at various angles of attack. The use of thrust-vectoring controls increased the maximum trim angle of attack from $55°$ (for the baseline configuration without thrust vectoring) to as high as $80°$. The results were expressed in terms of the maximum pitch rate achieved during these maneuvers,

and showed that even configurations with conventional thrust-to-weight ratios of about .7 could realize substantial increases in pitch-rate capability over an expanded angle-of-attack range compared with the baseline configuration without thrust-vectoring controls.

Simulation Fidelity

Historically, high-angle-of-attack simulations on the DMS have correlated well with flight tests, especially with respect to the identification of flight dynamics problems as well as airframe and flight control concepts to alleviate these problems. However, as was mentioned previously, the need for flight validation of the simulation fidelity has become apparent in recent airplane development efforts. In some aircraft programs significant discrepancies have been encountered between ground test facilities and between some ground test facilities and flight, as described in references 8 and 9. These experiences strongly suggest the need for flight validation to ensure confidence in ground-based results.

As was described previously, NASA is currently conducting full-scale flight tests of a research testbed F-18 known as the HARV, as part of the HATP program, in which the use of advanced controls for agility research and control margin/control law design criteria development methods are being investigated. The HARV is uniquely suited for high-angle-of-attack flight validation activities, as it is equipped for the monitoring of more than 700 flight test parameters and the use of flow visualization techniques. This flight test program will be used as an example to illustrate the process of assuring simulation fidelity. The approach to flight testing the HARV equipped with thrust-vectoring controls will be similar to other high-angle-of-attack flight tests which have been conducted. This approach includes updating the aerodynamic data base so that it consists of the best currently known information about the aerodynamics of the vehicle in order to validate the ability of ground-based simulations to predict reliably the dynamic response of the airplane to any pilot inputs. As flight data are becoming available at high angles of attack, parameter estimation efforts are under way to refine the aerodynamic data base for the

HARV. An important research objective of the HATP program is the improved modeling of aircraft dynamics at large angles of attack and sideslip.

One method of correlating large-amplitude simulation and flight motions is to compare numerical values of various figures of merit associated with such maneuvers. An example of this correlation between simulation and flight results for the HARV (without thrust-vectoring controls) is shown in figure 10. Shown is the time to roll through a bank angle change of 90° and the maximum roll rate achieved, starting from wings-level 1g trimmed flight and from M = 0.6 (accelerated conditions) with an initial bank angle of about 90°, versus angle of attack. Results for several maximum command roll maneuvers performed in flight tests are compared with the results obtained in the DMS simulation. These results indicate good correlation between simulation and flight. Additional correlation efforts are continuing which involve the use of parameter estimation techniques and non-real-time (batch) computer routines which use the recorded pilot control inputs or control surface deflections from flight to generate the resulting motions predicted by the simulation math model.

Evaluation Maneuvers

A fundamental test technique for high-angle-of-attack piloted simulation studies is the systematic progression, in distinct phases, from the performance of "open-loop" (i.e. pilot in the loop performing simple inputs) maneuvers to one-versus-one air combat, to one-versus-two engagements. Normally the first phase of an evaluation of a particular configuration with advanced controls involves pilot familiarization with the simulated airplane, evaluation of the high-angle-of-attack maneuvering characteristics of the airplane, and development of air combat maneuvering tasks for use in the next phase of the study. For studies to develop control margin design criteria, the primary evaluation maneuvers may be very few and "open loop", in order to focus on specific response characteristics for various levels of control effectiveness and to remove as many

control system effects as possible. Pilot familiarization of each configuration in such a study can be relatively brief.

The second phase of these piloted evaluations involves having the pilots fly the simulated airplane in closed-loop maneuvers. These maneuvers may involve the capture of a specific flight condition, flying against repeatable recorded air combat tasks, or engagements against a pilot in the other DMS dome. For agility research, this phase of the evaluation serves the purpose of quantifying the maneuvering benefits of advanced controls in realistic one-versus-one air combat situations and to uncover any handling qualities considerations or airframe/control system deficiencies which should be corrected. Agility and handling qualities research are closely related, as effective use of enhanced agility must be accompanied by acceptable handling qualities. Examples of studies which have specifically addressed handling qualities requirements at high angles of attack are described in reference 10. For control margin design criteria development, the performance of closed-loop and complex air combat maneuvering serves to validate or define any adjustments/refinements to the design criteria developed in the "open-loop" primary evaluation. The definition of control margin, agility, and handling qualities requirements determines the fundamental control law characteristics for enabling these requirements to be met. Unfortunately, a systematic, proven set of design guidelines and methodologies for the high-angle-of-attack control system development process to maximize agility and fully exploit high-angle-of-attack maneuvering capability does not yet exist. The HATP program is addressing this need.

Agility characteristics and design criteria must be evaluated under the most real-world conditions so that the complex maneuvers can be performed in rapid succession and the pilot's attention must be divided between flying the maneuvers, keeping track of a target, and managing a weapon system. Piloted simulation studies of one-versus-one air combat with one configuration having enhanced high-angle-of-attack agility and the other being a conventional fighter have been conducted for this purpose (refs. 11 and 12). Results from these investigations have shown large

benefits from the use of high-angle-of-attack agility. They have also quantified to some extent the level of benefit obtained from given amounts of control margin augmentation. The significant advantages seen in one-versus-one scenarios often come from the use of very high-angle-of-attack and low airspeed maneuvers. In an m-versus-n environment the level of augmentation required to obtain a significant advantage may be higher, and energy management will increase in importance. The next step in investigating high-angle-of-attack agility and design requirements is the simulation of one-versus-two engagements in which one highly agile vehicle engages two conventional configurations. The Highly Agile Vehicle Versus Two (HAVV TWO) program is currently under way at Langley to identify and evaluate additional considerations which the multi-bogie environment places on control effectiveness requirements and pilot situational-awareness needs. Some early results from this study are described in reference 13. This study began with very simple engagements and is progressing towards more complex engagements in order to enable quantification of the exchange ratio improvements due to enhanced agility and identification of the configuration characteristics that played a significant role in producing the improvements.

An important requirement for evaluation maneuvers used in piloted simulation studies is that they should relate as directly as possible to the airplane characteristics being evaluated, so that the pilot comments and ratings are meaningful and so that the quantitative results can be used as directly as possible. The maneuvers should be performed in a manner which insures that the critical flight conditions, pilot techniques, and resulting aircraft motions are examined. As was discussed previously, two significant high-angle-of-attack large-amplitude maneuvers are pure pitch maneuvers and rolls about the velocity vector. For high-angle-of-attack agility/advanced controls research, then, maneuvers which involve full pilot inputs in pitch and roll should be performed over the angle-of-attack and speed envelope of interest. The maneuvers should fully define the limits of the enhanced maneuvering envelope and agility/handling qualities design requirements and tradeoffs. These maneuvering characteristics can be defined by analyzing maneuvers in

which the pilot inputs are held until the maximum maneuvering rates are attained and those which involve closed-loop captures of specific conditions. These types of maneuvers should be performed in non-combat situations (for ease of analysis) as well as in tasks involving repeatable targets and in simulated air combat engagements. This approach also helps to identify any weaknesses or deficiencies in the control law design being used.

For some simulation research only one "open-loop" primary evaluation maneuver may be required. For control margin design criteria development, this approach allows many parametric variations of control effectiveness to be made and evaluated by several pilots. The initial conditions for evaluation maneuvers must also be carefully considered. For example, maneuvers used in the evaluation of control margin requirements, of which pilot ratings and comments on aircraft response may be an integral part, should be designed so that the motions that the pilot observes visually are generated only by the control moment capability of the airplane. Motions due to control system effects, thrust or other performance characteristics, or kinematic and other coupling motions should be minimized. By initiating such maneuvers at 1g stabilized trim conditions, at which there are no net forces or moments acting on the airplane such that $q = \dot{\alpha} = \dot{\gamma} = \ddot{h} = 0$, the thrust/performance effects are minimized. Figure 11 depicts this flight condition. The flight path angle (γ) will be less than zero (descending flight) at angles of attack where there is insufficient thrust to maintain level flight. These maneuver conditions are ideal for directly assessing the control moment available at that angle of attack. The primary maneuver used in the evaluation of nose-down pitch control requirements was a pushover from these conditions at a high angle of attack to low angles of attack. (See ref. 2.) A nose-down command applied at initial conditions at which the pitch attitude or the flight path angle is changing ($\dot{\theta}$ or $\dot{\gamma} \neq 0$) will result in changes in angle of attack that are not due solely to the nose-down moment generated by the application of nose-down controls. More complex maneuvering at a variety of flight conditions will be performed as part of the validation process in this study.

For purposes of quantifying and documenting the fundamental aircraft response characteristics and agility/maneuvering capabilities in a way which will be reproducible in flight tests for correlation with simulation results, the non-combat maneuvers performed in these evaluations should be repeatable and easily executable by the pilots. Maneuvers for which the initial conditions are dynamic (i.e. there are forces or moments acting on the airplane) will make the maneuver less repeatable, and will add complexity to the pilot technique if the timing of the pilot input is to be made at a specified point during the changing conditions. Pilot technique complexity is also increased if a sequence of inputs is required. Maximizing the repeatability and ease of execution of the maneuvers also minimizes difficulties in analyzing the results and comparing the results with full-scale flight motions. For closed-loop maneuvers in which flight conditions are captured within specified tolerances, these tolerances need to be tight enough to give meaning to the results and yet not so tight that they cannot be met in simulation and flight tests. In the progression from "open-loop" to closed-loop to air combat maneuvering, the repeatability and ease of pilot technique naturally decreases; however, by first obtaining a fundamental understanding of the results from simpler maneuvering, the analysis of more complex maneuvering will be simplified.

Role of Simulator Pilots in Evaluations

There are several factors which influence the effective use of research pilots for high-angle-of-attack simulation studies: (1) the number of participants and their backgrounds, (2) their involvement in the research process, (3) the establishment of their learning curves, (4) their acclimation to high-angle-of-attack motions, and (5) the approach taken to pilot ratings and comments.

It is highly desirable to use several research test pilots with extensive flight testing background from a variety of sources, including the military and industry. They should be familiar with air combat maneuvers employed with current fighter airplanes and should ideally be involved throughout the program. Any pilots who are involved in full-scale flight tests of a specific test configuration associated with the study will need to fly the simulator to obtain information prior to the test flights or to validate the simulation results if test flights have already been made.

Pilot involvement in the research process should begin with a thorough briefing regarding the background and purpose of the program and the simulator characteristics, if they are not familiar with them. They should be involved as much as possible in the development of the test techniques and the methodology to be used in the study, including the maneuvering techniques and assessment methods to be used.

An important aspect of the assessment method is the establishment of the learning curve before pilot comments are expressed or ratings are given. For simple, highly repeatable tasks, a particular configuration or parametric variation may be evaluated with very few runs; however, for more complex tasks in which the motions may vary due to the use of different pilot techniques, a number of runs may be required to establish the learning curve.

The performance of maneuvers at high angles of attack can produce unconventional motions which affect the pilot's perception of aircraft responses to his inputs. A primary example of this motion is the change in the aircraft's lateral-directional response to roll inputs at increasing angles of attack. Lateral inputs at high angles of attack to command a coordinated roll about the velocity vector produce an increasing proportion of body axis yaw rate compared with roll rate as the angle of attack is increased. When first encountered, this yawing motion can be disorienting or can appear to be a departure from controlled flight. Additional simulation time may be required for pilots to become acclimated to it. This phenomenon will be discussed further in a later section.

For some research in which specific pilot ratings are required in order to quantitatively document the pilot's opinion of an aircraft characteristic, existing accepted rating scales such as the Cooper-Harper handling qualities rating scale (see figure 12 and reference 14) may not be appropriate. A new rating

scale and/or rating approach may need to be developed. For instance, the Cooper-Harper scale is not applicable to piloted assessments of "open-loop" responses to simple inputs for which pilot compensation is not usually a factor, such as assessments of departure/spin recovery or rate capability. An example of a scale that was developed for the assessment of "open-loop" response to nose-down pitch commands is shown in figure 13 and is described in reference 2. The evaluation pilots were actively involved in the development and refinement of this scale, which has some structural similarity to the Cooper-Harper scale. In addition to the rating scale, a questionnaire which provided suggestions for qualitative pilot comments concerning additional pitch response characteristics and one which addressed the characteristics of the evaluation maneuvers were used and are shown in figures 14 and 15. These questionnaires were useful for generating additional pilot comments during the simulator sessions and debriefings. As a general practice for all piloted simulation studies, it has been found to be useful to obtain written summaries from pilots after each simulation session as further documentation and clarification of their evaluations.

Analysis of Results

The overall results of piloted simulation studies are generally derived from the analysis of aircraft motions and controls, pilot qualitative comments concerning workload and aircraft response, and quantitative pilot ratings. When qualitative or quantitative pilot opinion is used to make comparisons of maneuvering capability at different flight conditions or between aircraft configurations, it is desirable for them to be involved in the analysis process as much as possible in order to aid in the definition of the figures of merit which most influenced their opinions. The results should be expressed in terms of maneuvering performance and the effect of the variations which were made. For agility research, many figures of merit have been used and/or proposed to quantify the results of maneuvering capability. These figures of merit include the time to reach a flight condition or to capture it within a specified tolerance, and maximum angular changes, rates, or accelerations achieved during the maneuver. As yet, there is no generally accepted

specific set of figures of merit (also referred to as metrics) for quantifying high-angle-of-attack agility. A sample presentation of the results for simulated "open-loop" roll maneuvers was shown in figure 10. These results for the F-18 HARV with and without thrust-vectoring controls are shown in figure 16. The results show that the two figures of merit used are clearly useful for quantifying the enhanced roll agility achieved with the use of thrust vectoring. Such results can also be used to define control law design goals.

A number of ways to meaningfully quantify maneuvering enhancements in simulated air combat engagements also exist. In particular, such overall figures of merit as the angle (ϵ) between the aircraft X body axis and the range vector to the opponent and the rate of change of this angle indicate instantaneous maneuvering advantage. The time on advantage, defined as the cumulative time during which the aircraft $\epsilon < 90°$ and the opponent's $\epsilon > 90°$ is an indicator of sustained maneuvering advantage. The results as indicated by these and other measures of maneuvering advantages during air combat should be expected based on an understanding gained from earlier analysis of non-combat maneuvers. Of course, the victor in any air combat engagement will be the first one who satisfies the weapons firing/launching parameters, which normally involve ϵ, the range between aircraft, and other requirements. By performing sufficient numbers of engagements, a meaningful probability of a specific configuration being the victor against some other configuration can be determined. References 11 through 13 contain analyses of combat maneuvers and engagements for configurations with and without thrust-vectoring controls.

A particular data analysis process is appropriate for the determination of control margin design guidelines involving pilot ratings. The determination of appropriate candidate figures of merit for the analysis will be discussed in a later section; however, for each candidate figure of merit selected, the level of statistical correlation should be determined between the quantitative values for that response characteristic and the pilot comments and ratings assigned. In this manner the most significant figure(s) of merit that best characterize those aspects of the

response that the pilots evaluated can be determined. This process is depicted in figure 17. The statistical correlation method that was found to work well for the determination of the figures of merit for nose-down control response was to compute the mean values of the figure of merit versus pilot rating and the 95-percent confidence intervals about the mean at each rating value. For this study, one figure of merit that was determined to be significant was the maximum nose-down pitch acceleration achieved within the first second of a full nose-down command at high angles of attack. These results are shown in figure 18. The clear dependence of pilot rating on the amount of pitch acceleration achieved and the generally small confidence intervals were evidence of a meaningful correlation.

FLIGHT VALIDATION OF RESULTS

As was shown in figure 4, final determination of the results of high-angle-of-attack piloted simulation studies involves the use of ground-based testing and full-scale flight testing to validate the simulation results. These tests are used to determine any refinements needed to the simulation mathematical model or the evaluation methodology used, such as the maneuvers and rating approaches. The simulation also serves as a tool for flight test planning and practice for the test pilots. In flight tests, real-world considerations with respect to pilot/vehicle-interface needs can be evaluated and their effect on the validity of the simulation results assessed. These considerations include cockpit displays and controls as well as motion/physiological effects such as spatial disorientation and accelerations experienced by the pilot. The following sections will discuss the use of flight testing for the validation of simulation results.

Validation of Maneuvers and Rating Approaches

Considerations. - An evaluation of the maneuvers performed and the validity of the rating approaches used in the simulation studies must also be made in flight tests. It is important that the simulation results be based on realistic maneuvers that can be performed in flight within acceptable tolerances for the maneuver performance without violating any aircraft restrictions or requiring excessive pilot workload. As

an example, during full-input large-amplitude rolls at high angles of attack, holding the angle of attack nearly constant during the maneuver can be a high workload task, both in simulation and flight tests. If an additional requirement such as capturing a roll angle is added, the workload may be unacceptably high, especially if the tolerances are tight and the handling qualities are poor. The simulation results should also accurately predict the pilot's qualitative opinion and numerical ratings in full-scale flight. If the pilot's opinion of the aircraft response is significantly affected in flight due to factors such as the effects of motion, the fixed-based simulation results will need to be modified. It may also be determined in flight tests that the pilot rating approach itself needs to be altered. Future flight tests of the HARV will yield such information concerning the validity of the nose-down control margin study conducted on the DMS and the application of the Cooper-Harper handling qualities rating scale to enhanced high-angle-of-attack flight.

Status of Maneuver Definition. - During flight test programs, as the airplane is cleared for different regions of the flight envelope from benign flight conditions to more demanding ones, maneuvers and tests performed during piloted simulation are repeated for evaluation/validation purposes. Accepted task performance guidelines for nonlinear piloted simulation of high-angle-of-attack maneuvering and corresponding evaluation procedures/guidelines for flight test do not currently exist. Historically, different ad hoc approaches have been used by various organizations during specific programs. However, little attempt has been made to pull together these various approaches and take advantage of the lessons learned over the years. Therefore, development of open- and closed-loop task performance guidelines and evaluation procedures that are generally accepted for agility research and control law evaluations is a current and future research challenge.

High-angle-of-attack research programs are attempting to address the issue of task definition. NASA has proposed that a set of standard, representative tasks be defined and used in all ongoing high-angle-of-attack research flight programs, with the same tasks being evaluated in simulation and flight. Still unresolved is what the specific tasks should be.

Research activities are underway to develop and ultimately flight-validate candidate tasks. Starting with the fundamental characteristic maneuvers shown in figure 7 as a basis, a preliminary set of candidate maneuvers which could be used for high-angle-of-attack agility and control law design research is being evaluated using the DMS. Figure 19 describes this candidate set of maneuvers, which is designed to evaluate the aircraft's ability to rapidly point the nose relative to the flight path, as depicted in figure 1. In addition to these nose-pointing maneuvers, others are being developed on the DMS which will demonstrate the second aspect of agility shown in the figure -- the ability to reposition the aircraft by quickly turning the velocity vector. The DMS is also being used to develop flight test maneuvers based on the handling qualities evaluations described in reference 10. Flight tests of these various types of maneuvers using the F-18 HARV will validate their utility for high-angle-of-attack agility, handling qualities, and control law evaluations.

Pilot/Vehicle Interface Considerations

An important goal of the research within the HATP program is to define the considerations and needs of the pilot with respect to cockpit displays and controls, the possibility and consequences of spatial disorientation during maneuvers, and the severity and effect of g loads experienced by the pilot. Any or all of the potential difficulties associated with these factors can affect the validation of piloted simulation results because they can cause problems with the accuracy and repeatability of the test points.

The presentation of cockpit displays and the mechanization of the controls can affect both the pilot's ability to perform a maneuver and his opinion of the aircraft response. For example, if a display of critical information for performance of the maneuver is difficult to read because of its design or placement, the maneuver performance and/or pilot opinion may be affected. Because of the nature of high-angle-of-attack flight and potential problems with spatial disorientation, the performance of large-amplitude maneuvers at these conditions may require the use of unconventional

displays. The DMS is being used to investigate the use of helmet-mounted displays with a view towards flight tests of such a system on the F-18 HARV.

Spatial disorientation, which can cause the maneuver performance to suffer due to a reduction in situational awareness, can result due to the occurrence of unusual flight attitudes or motions. Such disorientation can occur within the conventional flight envelope at angles of attack below the stall; however, the possibility exists, based on simulation experience, that more severe disorientation may result during the performance of maneuvers such as high-angle-of-attack rolls about the velocity vector. Pilots who are used to rolling about the longitudinal body axis at low angles of attack may become very disconcerted at first by the substantial initial yawing motion observed in response to a roll input. Pilots have adapted to this phenomenon in simulations; however, there is currently a lack of flight experience with these motions. Simulation results related to agility and design criteria development may have to be altered after comparing these results with flight test data. Applicable data should be available soon from the F-18 HARV flight tests and other high-angle-of-attack flight programs. This issue is also being addressed as part of the research being conducted with helmet-mounted displays mentioned previously.

A second physiological consideration for pilot/vehicle interfacing is the potential for excessive accelerations (g loads) encountered at the pilot station during rapid maneuvering at high angles of attack. The primary concerns are the onset rate of normal acceleration on the pilot during rapid pitch maneuvers, the buildup of axial acceleration ("eyeballs-out" g's) due to high yaw rates during high-angle-of-attack rolls, and the lateral accelerations experienced due to rapid yaw accelerations in these rolls. These values can be easily calculated from simulation data; however, only flight tests will determine exactly how the pilots will respond to these motions and how much they will affect the results of simulation studies.

CONSIDERATIONS FOR DESIGN CRITERIA
DEVELOPMENT

In addition to the test techniques/methodology and validation considerations previously described, there are three aspects of piloted simulation studies which should be included in the development process for high-angle-of-attack design criteria: (1) the evaluation of candidate criteria, (2) the relationship of these criteria to the design process, and (3) the specification of requirements for demonstrating in flight that the design criteria have been met. The overall process being used in the development of nose-down pitch control margin design criteria is shown in figure 20.

Evaluation of Candidate Criteria

Any study to define design criteria should include the evaluation of candidate criteria, beginning with a review of any available literature for existing or proposed criteria or guidelines, in order to determine how much work has been done, how systematic and comprehensive the work was, and how well the results agree with each other. If a reasonable data base of simulation and flight test results exists, sufficient information may be obtained to define a preliminary guideline which can be compared with the simulation results at the completion of the study. The results of such a review of existing guidelines and data bases for nose-down pitch control criteria are contained in reference 15.

An important step in the evaluation of control margin requirements which relates to the use of pilot rating approaches as well as the analysis of the simulation results for the development of design criteria is the establishment of figures of merit to be used in evaluating the aircraft response. As many potential figures of merit as possible should be considered. They can best be compared by characterizing them according to the strength of their relationship to control power and the time scale relative to initiation of the pilot command. Figure 21 shows this overall relationship for a number of potential figures of merit which were considered for nose-down pitch control capability. Clearly, in the absence of significant angular rates, pitch acceleration ($\dot{q}$) bears a strong relationship to pitch

control power because it is directly proportional to static pitching moment coefficient (C_m). The longer time-scale parameters shown on the right end of the plot have a much weaker association with control power and are more closely associated with airplane performance effects such as thrust and drag. Therefore, those figures of merit on the left side of the scale would be expected to be the more critical ones for nose-down control design consideration, although the others could also be useful as supplemental or check parameters.

A final area of consideration for the evaluation of candidate design criteria is the generation of a systematic, comprehensive data base of simulation results, from which the final criteria can be derived. The performance of sufficient runs to ensure the establishment of the pilots' learning curves and a statistically meaningful set of results was discussed previously. For control margin design these results should also incorporate the variation of critical parameters affecting control capability and response. For example, those parameters which were chosen to characterize the static nose-down pitching moment characteristics are illustrated in figure 22 and include: (1) the minimum value of C_m, $C_m{}^*$, (2) the angle-of-attack range over which $C_m{}^*$ occurs, $\Delta\alpha^*$, and (3) the slopes of the pitching moment curve for angles of attack below and above $\Delta\alpha^*$, S1 and S2. Such parameters should be varied individually and systematically for the piloted evaluations. For the nose-down control margin study, 25 separate parametric variations of the nose-down pitching moment capability were evaluated. The range of variations for each characteristic evaluated were based on the characteristics of current aircraft and projected future designs. As a preliminary check on the validity of the initial quantitative analysis of the simulation results, additional maneuvers were performed to verify that the pilot ratings could be predicted for a wide variety of control margin characteristics.

Relationship of Simulation-Derived Criteria to the Design Process

During the early design stages of a new aircraft, the aircraft designer requires guidelines which enable him to design for the desired aircraft performance. Ideally it is best to apply design guidelines as early as possible in the design process such that significant design problems can be identified and design tradeoff studies can be conducted. The format of design guidelines must be easy to apply and yet comprehensive in including the most significant factors which influence the performance. An example of the early application of design guidelines is during preliminary wind-tunnel screening of candidate configurations in which quick assessments of stability levels and control effectiveness are made.

Criteria developed from piloted simulation can be very useful in developing design guidelines. Usually the intent will be that the aircraft achieve the desired performance demonstrated as satisfactory in the simulator. Very importantly, the designer must have a high level of confidence that use of simulation-derived guidelines will ultimately produce aircraft which meet the original criteria. To achieve this high level of confidence the design guideline must capture the intent of the criteria, including pilot opinion, and ideally should be flight-validated on a variety of aircraft.

A design guideline was developed from nose-down pitch control margin simulation results reported in reference 2. This guideline provided a methodology to determine the minimum value of C_m required at the pinch point (C_m^*) and the shape of the available nose-down pitch response. The basis for this guideline included considerations for pitch acceleration and pitch rate requirements during 1g pushover maneuvers, and the guideline is illustrated in figure 23. In addition the methodology to account for inertia coupling increments during rolling maneuvers was also developed. The process to select the pitching moment required at each angle of attack including inertia coupling considerations is illustrated in figure 24.

Flight Test Demonstration Requirements

In the final stages of the design process for a new configuration, flight test is used to demonstrate that the configuration meets design requirements and/or is in compliance with the specifications. Typically a comprehensive set of flight demonstration requirements is outlined prior to the flight test phase and is methodically completed as the flight envelope is expanded.

Piloted simulation is very useful for developing flight demonstration requirements, especially those which are related to simulation-derived design criteria. The specific test techniques and flight conditions can be developed in the simulator in order to determine optimum piloting techniques and the most efficient methods for acquiring the demonstration data. Specific test conditions which are difficult to achieve or assess can be identified prior to flight test. Also, operational constraints on flight demonstrations can be evaluated and alternative demonstration requirements can be developed when required.

Flight demonstration maneuvers which are used to demonstrate design criteria ideally should be closely related to maneuvers used in simulation to develop the criteria. This approach allows the fundamental understanding of the flight dynamics gained from simulation to be applied to flight test and assures that the design methodology is reflected in the maneuver requirements. The flight demonstration should be repeatable and easily accomplished using normal flight testing techniques and not require unusual flight instrumentation for data documentation.

The DMS was used to develop flight demonstration requirements for the nose-down pitch control criteria as previously discussed. The recommended flight maneuvers were closely related to the basic criteria development maneuvers used in simulation. These maneuvers included stabilized 1g pushovers, pushovers during rolling maneuvers, pull-push and zoom climb maneuvers. Successful demonstration of meeting the design criteria included achieving threshold values of pitch acceleration and pitch rate within specified time periods. The DMS was

349

particularly useful for developing the specific test techniques for the flight demonstration. Techniques for achieving 1g stabilized conditions at high angles of attack were evaluated including initial conditions, stabilization criteria, and the impact of engine operating limitations. Also, maneuvers were developed to demonstrate nose-down pitch control during rolling maneuvers which were very complex and difficult to evaluate. An understanding of the flight mechanics associated with recoveries from zoom climbs was also achieved. In summary, piloted simulation using the DMS proved to be invaluable in developing maneuvers which would safely and efficiently demonstrate compliance with these design requirements. As a final step, these simulation-derived maneuvers will be evaluated in a flight validation program using the F-18 HARV.

CONCLUDING REMARKS

Piloted simulation has been an important tool for high-performance aircraft flight dynamics research at NASA-Langley. It has a major role in the high-angle-of-attack technology development process in the NASA High-Angle-of-Attack Technology Program (HATP), particularly for agility research and design criteria development. The Differential Maneuvering Simulator is the primary facility used for this research, and has been used as an effective research tool to develop the design methodologies required to implement advanced technologies on future aircraft.

Test techniques and methodologies have been developed for the effective use of the simulation capabilities. Software requirements, particularly the high-angle-of-attack math modeling of the aerodynamic characteristics, are critical to the successful application of the simulation results. Evaluation maneuvers which are repeatable, easy to execute, and relevant to the research objectives are developed and are usually performed in a progression from the most simple to complex maneuvering and air combat engagements. The most effective use of simulator pilots requires their participation in the research process, particularly for the development of maneuvers and rating approaches and for the identification of appropriate figures of merit for analysis of the results. The data base generated should

reflect the establishment of the pilots' learning curves and for control margin design criteria development have statistical significance.

Correlation with full-scale flight results is the primary means of validating the simulation results and approach. The fidelity of the simulation math model can be verified by comparing flight and simulation motions. The utility of evaluation maneuvers and pilot rating approaches used in simulation can be examined in flight. Pilot/vehicle interface considerations and their impact on the simulation results can also be assessed.

In order to develop design criteria, additional steps are required in the simulation study approach. Candidate design criteria must be carefully evaluated and a systematic, comprehensive data base of simulation results generated. The final criteria developed must be easily applicable to the design process and successfully predict the aircraft performance. Piloted simulation can be used to define flight test demonstration requirements, which can be evaluated in full-scale flight tests.

REFERENCES

1. Gilbert, William P.; Nguyen, Luat T.; and Gera, Joseph: Control Research in the NASA High-Alpha Technology Program. AGARD CP-465, October 1989.

2. Ogburn, Marilyn E.; et. al.: High-Angle-of-Attack Nose-Down Pitch Control Requirements for Relaxed Static Stability Combat Aircraft. Paper presented at NASA High-Angle-of-Attack Technology Conference, Hampton, Virginia, October 30-November 1, 1990.

3. Chambers, Joseph R.; Gilbert, William P.; and Nguyen, Luat T.: Results of Piloted Simulator Studies of Fighter Aircraft at High Angles of Attack. Paper presented at AGARD Fluid Dynamics Panel Symposium on Dynamic Stability Parameters, Athens, Greece, May 1978.

4. Ashworth, B. R.; and Kahlbaum, William M., Jr.: Description and Performance of the Langley Differential Maneuvering Simulator. NASA TN D-7304, June 1973.

5. Gilbert, William P.; and Nguyen, Luat T.: Use of Piloted Simulation for Studies of Fighter Departure/Spin Susceptibility. Paper presented at AGARD Flight Mechanics Panel Meeting on Piloted Aircraft Environment Simulation Techniques, Brussels, Belgium, April 1978.

6. Nguyen, Luat T.; et. al.: Simulator Study of the Stall/Post-Stall Characteristics of a Fighter Airplane with Relaxed Longitudinal Static Stability. NASA TP-1538, December 1979.

7. Ogburn, Marilyn E.; Nguyen, Luat T.; and Hoffler, Keith D.: Modeling of Large-Amplitude High-Angle-of-Attack Maneuvers. AIAA 88-4357-CP, August 1988.

8. Hammett, L. N., Jr.: An investigation of the F-16 High-Angle-of-Attack Pitching-Moment Discrepancy. Technical Report AFWAL-TR-81-3107.

9. Erickson, Gary E.: Water Tunnel Flow Visualization and Wind-Tunnel Data Analysis of the F/A-18. NASA CR 165859, May 1982.

10. Wilson, David J.; and Riley, David R.: Flying Qualities Criteria Development Through Manned Simulation for 45° Angle of Attack. NASA CR 4311, July 1990.

11. Ogburn, Marilyn E.; et. al.: Simulation Study of Flight Dynamics of a Fighter Configuration with Thrust-Vectoring Controls at Low Speeds and High Angles of Attack. NASA TP 2750, March 1988.

12. Doane, P. M.; Gay, C. H.; Fligg, J. A.; et.al: Multi-System Integrated Control (MuSIC) Program. WRDC-TR-90-6001, June 1990.

13. Hoffler, Keith D.; et. al.: Utilization and Benefits of Advanced Aerodynamic and Propulsive Controls: A Simulator Study. Paper presented at NASA High-Angle-of-Attack Technology Conference, Hampton, Virginia, October 30-November 1, 1990.

14. Cooper, George E.; and Harper, Robert P., Jr.: The Use of Pilot Rating in the Evaluation of Aircraft Handling Qualities. NASA TN D-5153, 1969.

15. Nguyen, Luat T.; and Foster, John V.: Development of a Preliminary High-Angle-of-Attack Nose-Down Pitch Control Requirement for High-Performance Aircraft. NASA TM-101684, February 1990.

Point and shoot   **Positioning**
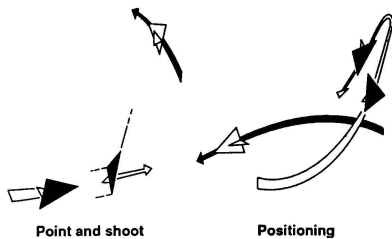
Figure 1. - Illustrations of high-angle-of-attack agility
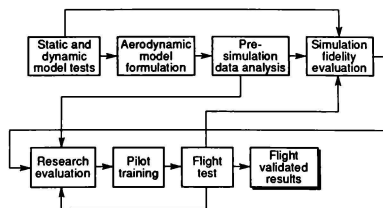
Figure 4. - Approach used for piloted simulation studies
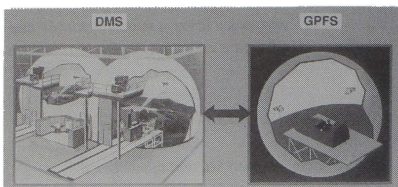
Figure 2. - Simulator facilities used for one-versus-two air combat studies

Figure 3. - Piloted simulation agility research

Figure 5. - View of DMS cockpit and visual display
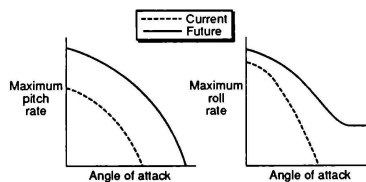
Figure 6. - Comparison of current with future pitch and roll rate capability
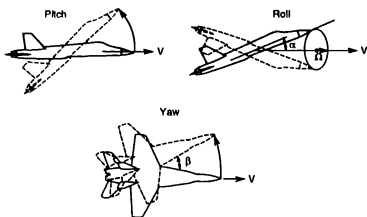
Figure 7. - Characteristic maneuvers

Figure 9. - Maximum pitch rates achieved from 1g trim conditions for various levels of T/W for a configuration with thrust vectoring. $h_o = 20,000$ ft.
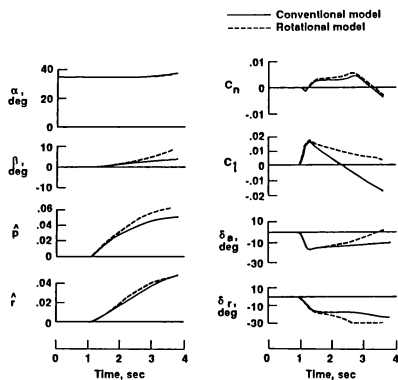
Figure 8. - Time history of large amplitude roll maneuver at $\alpha_o = 35°$ with and without rotary aerodynamics modeled

Figure 10. - Comparison of roll maneuvering results for simulation and flight. $h_o = 25,000$ ft.

Figure 11. - Stabilized 1g trimmed conditions

Figure 12 (Cooper-Harper handling qualities rating scale):

Adequacy for Selected Task or Required Operation*	Aircraft Characteristics	Demands on the Pilot In Selected Task or Required Operation *	Pilot Rating
	Excellent Highly desirable	Pilot compensation not a factor for desired performance	1
	Good Negligible deficiencies	Pilot compensation not a factor for desired performance	2
	Fair Some mildly unpleasant deficiencies	Minimal pilot compensation required for desired performance	3
	Minor but annoying deficiencies	Desired performance requires moderate pilot compensation	4
	Moderately objectionable deficiencies	Adequate performance requires considerable pilot compensation	5
	Very objectionable but tolerable deficiencies	Adequate performance requires extensive pilot compensation	6
	Major deficiencies	Adequate performance not attainable with maximum tolerable pilot compensation. Controllability not in question	7
	Major deficiencies	Considerable pilot compensation is required for control	8
	Major deficiencies	Intense pilot compensation is required to retain control	9
	Major deficiencies	Control will be lost during some portion of required operation	1 0

Decision path:
- Is it satisfactory without improvement? — YES → ; NO → Deficiencies warrant improvement
- Is adequate performance attainable with a tolerable pilot workload? — YES → ; NO → Deficiencies require improvement
- Is it controllable? — YES → ; NO → Improvement mandatory
- Pilot decisions

* Definition of required operation involves designation of flight phase and/or subphases with accompanying conditions.

Figure 12.- Cooper-Harper handling qualities rating scale

Figure 13 (Pitch recovery rating scale):

Control Margin Adequacy	Pitch Response Characteristics	Mission Suitability/ Safety	Pilot Rating
	Excellent pitch response	Enhancing for mission	1
	Desirable recovery (recovery not a concern)	Satisfactory for mission	2
	Adequate recovery	Degrades mission slightly	3
	Recovery was not in doubt but marginal	Moderately degrades mission	4
	Recovery was in doubt and/or required unnatural control inputs	Significantly degrades mission and/or safety	5
	Insufficient nose-down pitch response results in out-of-control flight	Unacceptable for mission and safety	6

Decision path:
- Is the recovery highly desirable? — Yes → ; No → Nose-down pitch control margin is satisfactory
- Is the recovery adequate? — Yes → ; No → More nose-down pitch control margin is required
- Is the aircraft recoverable? — Yes → ; No →
- Pilot decisions

Level A: Tactically desirable — 2.5
Level B: Minimum for non-tactical safety — 4.5

Decision Factors
1. Was there enough pitch response (acceleration, rate)?
2. Could you use more response?
3. Was time to recover short enough?
4. Was the recovery in question?
5. Was pilot compensation required?
6. Is the response suitable for the mission?

Figure 13. - Pitch recovery rating scale

Use the following questions as a guideline for describing and evaluating each test point.

1. **Describe respone to stick input.**
   a. Pitch response
   b. Accompanying roll/yaw motions
   c. Disorienting motion

2. **Compare this response to other aircraft you have flown.**
   a. Aircraft
   b. Conditions
   c. Similar, better or worse

3. **Give your opinion on the application of this maneuver to combat.**
   a. Characteristics that enhance or degrade combat effectiveness
   b. Describe what you would most like to improve on this response

4. **Determine impact of other influences on your opinion.**
   a. Did recovery time affect your opinion?
   b. Did altitude loss affect your opinion of the recovery?
   c. Were you most concerned about mission safety or mission accomplishment during this maneuver?
   d. Did pilot technique affect results?
   e. What pilot compensation was required to complete maneuver?

Figure 14. - Pilot questionnaire for additional comments

Use the following questions as a guideline for describing the potential tactical applications of each type of maneuver.

1. Based on your experience, would this maneuver be tactically useful for current or future aircraft?

2. What would you like to improve on this maneuver to increase the tactical effectiveness?

3. If this aircraft displayed "excellent" response capability, would this maneuver be tactically useful?

4. Would you desire to have more AOA capability than that demonstrated during this maneuver and why?

5. Describe a tactical situation where you would most likely see this setup and desire to perform this maneuver.

6. What maneuver/s would likely precede and follow this in a tactical situation?

Figure 15. - Pilot questionnaire for evaluation of maneuvers
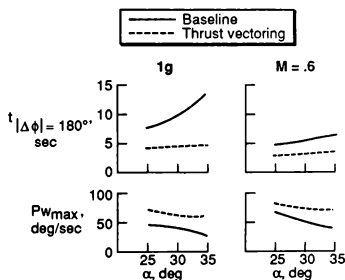
Figure 16. - Roll maneuvering results from simulation evaluation. $h_o = 25,000$ ft.
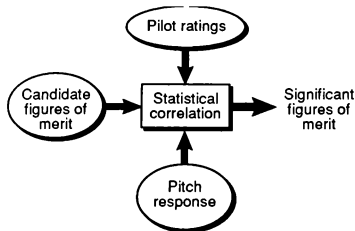
Figure 17. - Approach to data analysis for control margin studies
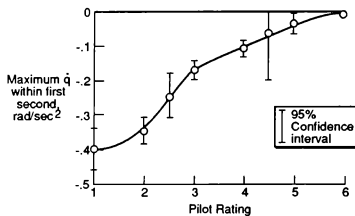
Figure 18. - Variation of maximum $\dot{q}$ achieved within one second of recovery initiation with pilot rating

Nose-up : Max positive pitch rate
  α capture
  θ capture

Nose-down : Max negative pitch rate
  α = 0° capture
  θ = 0° capture

Max rate roll through
  $|\Delta\phi_w| = 180°$

$|\Delta\phi_w| = 90°, 180°$ capture
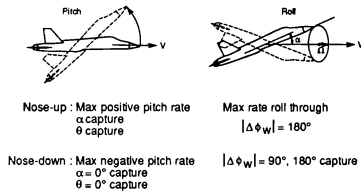
Figure 19. - Candidate maneuvers for agility and control law evaulations
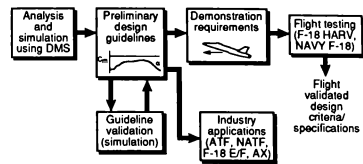
Figure 20. - High-alpha nose-down pitch control margin requirements program
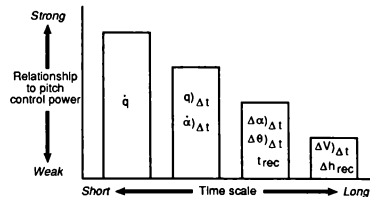
Figure 21. - Candidate figures of merit
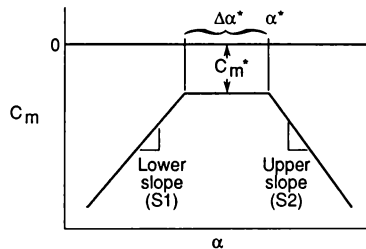
Figure 22. - Parametric variations of nose-down pitching moment used in simulation study

356
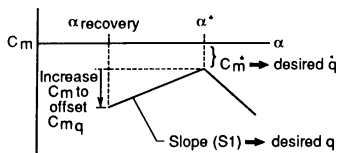
Figure 23. - Illustration of C_m design guidelines for
nose-down pitch response
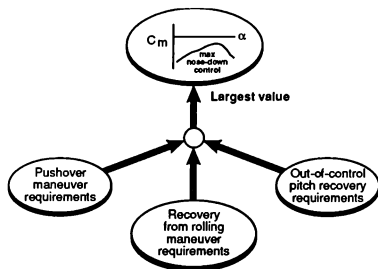
Figure 24. - Illustration of determination of required C_m
at each angle of attack

UTILIZATION OF SIMULATION TOOLS IN THE
HL-20 CONCEPTUAL DESIGN PROCESS

E. Bruce Jackson*
NASA Langley Research Center
Hampton, Virginia 23665

W. A. Ragsdale†
UNISYS Corporation, Hampton, Virginia
Hampton, Virginia 23665

Richard W. Powell**
NASA Langley Research Center
Hampton, Virginia 23665

Abstract

The NASA Langley Research Center (LaRC) is in the
conceptual design stage of the Personnel Launch System
(PLS). The passenger-carrying portion of the PLS system is
a 20,000 pound lifting body vehicle, known as the HL-20.

Previous programs have demonstrated the controllability
and "landability" of unpowered lifting body vehicles with
low lift-to-drag (L/D) ratios; however, one of the early
lifting body designs demonstrated an instability in the
lateral-directional axes.

To evaluate the flight characteristics of the HL-20 design, a
real-time simulation of the HL-20 lifting body vehicle has
been developed at LaRC. The simulation model is being
used to validate the HL-20 concept, identify opportunities
for improving the design of the vehicle, and to develop
preliminary designs for both automatic and manual control
systems.

This paper describes the development of the real-time
simulation, including development of manual and automatic
flight control laws. Key results from use of this simulation
are described, including identification of improved landing
gear geometry, a requirement for aerodynamic
improvements, and increased confidence in the improved
HL-20 design.

Introduction

The NASA Langley Research Center (LaRC) is in the
conceptual design stage of the Personnel Launch System
(PLS). The Personnel Launch System consists of a booster
and a lifting body, coupled with adapter hardware, to allow
for vertical launch to orbit, followed by an unpowered
horizontal landing at the end of the mission.   The
passenger-carrying portion of the PLS system is a 20,000
pound lifting body vehicle, known as the HL-20.   The
HL-20 lifting body design features a crew compartment
large enough to house two crewmembers and eight
passengers (figure 1).

The HL-20 is intended to serve as an Assured Crew Return
Vehicle for Space Station Freedom, as a "space taxi" to
ferry astronauts to and from the space station, and as a

vehicle to perform other low earth orbit missions that do not
require significant payload capability (satellite repair, free-
flyer platform maintenance, and orbital rescue are
examples). Other design features include a detachable one-
piece heat shield, easy access to vehicle subsystems via lift-
off panels for maintenance, and rapid mission turn-around
following a horizontal landing. The cross-range capability
is sufficient to allow a daylight landing at one of five
designated landing sites on any orbit.

Previous programs[1,2,3] have demonstrated the
controllability and "landability" of lifting-body vehicles
with low lift-to-drag (L/D) ratios; however, an early lifting-
body demonstrated an instability in the lateral-directional
axes.[4]

Early identification of problems in the flight characteristics
of the HL-20 should result in less expensive solutions than
if the problems are discovered later in the development
program.

In order to identify deficiencies in the HL-20 concept,
simulation studies of the flight characteristics were
developed, and several candidate control laws were
designed. These simulation studies included a nonreal-time
simulation, used for launch, orbit, and re-entry and
guidance to final approach to the landing facility, and a
real-time simulation used to study the low supersonic and
subsonic phases of the re-entry, including approach and
landing on a simulated runway.

An earlier paper described the nonreal-time simulation
study and results.[5] This paper will describe the real-time
simulation setup, the HL-20 real-time simulation model,
and several candidate flight control system designs, as well
as preliminary results from these real-time studies.
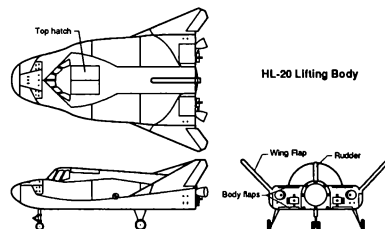
Figure 1. - HL-20 Three View Drawing

*Aerospace Engineer, Aircraft Guidance and Controls
Branch, Guidance and Control Division, Member AIAA

**Aerospace Engineer, Vehicle Analysis Branch, Space
Systems Division, Member AIAA

†Staff Engineer, Senior Member AIAA

D   Drag, lbs

g   Gravitational constant (approx. 32.2 ft/sec^2)

L   Lift, lbs

N_Z Normal acceleration, g's

$\dot{V}$ Rate of change of velocity, ft/sec^2

V_{rel} Velocity relative to earth, ft/sec

γ Flight Path Angle, degrees

$\dot{\gamma}$ Flight Path Angle rate, degrees/second
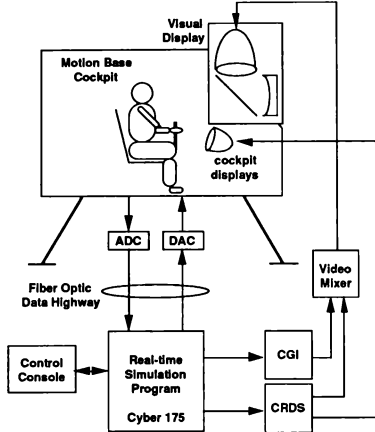
Figure 2. - Real Time Simulation Schematic

Figure 3. - Electronic Attitude Display Indicator Schematic
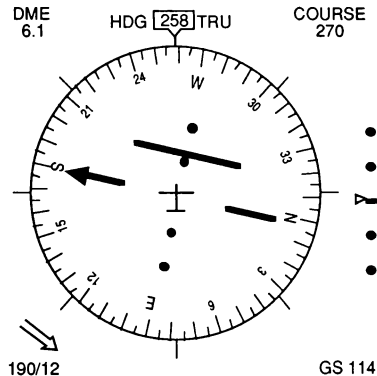
Figure 4. - Horizontal Situation Display Schematic

Discussion

Real-time Simulation Set-up

To find out if the HL-20 concept had any serious deficiencies in performance or handling qualities during the approach and landing phase, a real-time simulation effort was undertaken. Using preliminary wind tunnel data, a subsonic aerodynamic model of the HL-20 was developed[6]. This aerodynamic model was combined with an atmosphere model, equations of motion, and preliminary control laws to yield a six-degree-of-freedom approach and landing simulation capability. This model was then installed on the Transport Systems Research Vehicle fixed-base cockpit at LaRC, and used to conduct preliminary flying qualities and landing studies[7]. Follow-on studies have been performed using the Visual Motion Simulator cockpit, with an expanded aerodynamic model valid to Mach 4.

The host computer used in these studies was a Control Data Corporation CYBER-175, running at a frame rate of 33.3 Hz. Simulation peripheral equipment in both cases included an Evans and Sutherland CT-6 computer image generator (CGI), and a McFadden side-arm controller. The heads-down display was provided by a Terabit Eagle graphics generator, known as the calligraphic/raster display system (CRDS). For further realism, a sound system provided generic wind noise and landing gear touchdown sounds. A diagram of the setup used in these simulations is shown in figure 2. Representations of the cockpit electronic attitude display indicator (EADI) and the horizontal situation display (HSD) are shown in figures 3 and 4.

A heads-up display (HUD) was provided, in later studies, by the CRDS. The HUD included flight director and velocity vector information, as well as airspeed, altitude, flight path reference markers and a preflare cue, in a manner similar to one of the declutter modes of the Shuttle Orbiter HUD. A typical HUD representation is shown in figure 5.
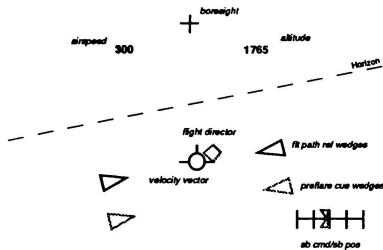
Figure 5. - Heads-Up Display schematic

A set of Precision Approach Path Indicator (PAPI) lights were displaced on the extended centerline of the runway to serve as the aimpoint for the outer glideslope. Unlike the Shuttle Landing Facility, however, no inner glideslope (ball-bar) was depicted. A typical view as seen by the pilot through the forward window while on approach to landing, including overlaid HUD symbology, is shown in figure 6.
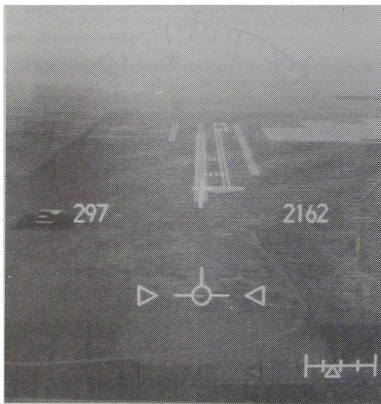
Figure 6. - Pilot's view approaching preflare height

Flight Control Systems

Several different flight control systems were designed to provide artificial stabilization and flight control of the vehicle in the subsonic regime. These control systems included a set of candidate manual control systems for use in the subsonic regime, and an automatic landing system. These control systems were developed using both real- and nonreal-time simulation facilities.

Baseline Control Laws

The initial control law, used in early studies, included a simple rate feedback system which provided increased damping in all three axes, and included an aileron-to-rudder interconnection to improve the coordination of turns (figure 7). It was later discovered that this interconnection
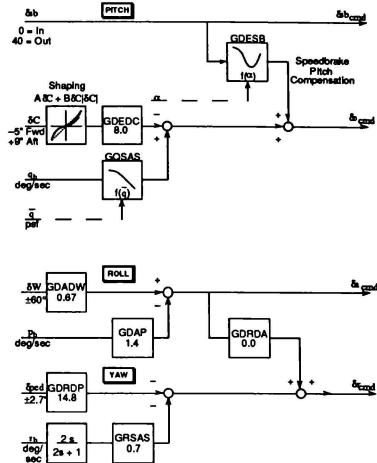
Figure 7. - Baseline Flight Controls

caused difficulties in performing uncoordinated maneuvers during crosswind landings and was removed with minimal impact on sideslip response in turns.

As shown in figure 7, the pilot's pitch input (delta column, or δC) passes through a shaping filter, then multiplied by a gain (GDEDC). The sign is then reversed to match elevator deflection sign convention (trailing edge down is positive). For stability, the aircraft pitch rate (q_b) is fed back at this point. A portion of the speedbrake command (δsb) is added to the result to remove pitch response from speedbrake deflection. The sum of these three signals is sent to the control surface mixer as δe_{cmd}.

The roll axis combines the pilot's roll input (delta wheel, or δW) with a roll rate feedback measurement (p_b) for stability. This signal is then sent to the control surface mixer as an "aileron" command (δa_{cmd}).

The pilot's rudder pedal position (δped) is summed with yaw rate feedback for stability. The yaw rate signal (r_b) is first sent through a washout filter to allow for steady turns. A portion of the aileron command is added to this signal when gain GDRDA is non-zero; simulator studies showed better crosswind landing capability when GDRDA was zero. The result is sent to the control surface mixer as δr_{cmd}.

In the course of piloted simulator evaluations of the HL-20, several modifications to the initial control system were made to make the flared, unpowered landing easier to accomplish. These improvements were made only to the pitch axis; the lateral/directional control laws remained unchanged. These "experimental" control laws are described below.

Experimental Control Laws - Justification

Landing a low-L/D vehicle, such as the HL-20, is very different from landing a conventional aircraft. The sink rate

on final approach is over 150 ft/sec at a speed of about 300 knots. This must be reduced to less than 5 ft/sec as the aircraft approaches the ground, and maintained at a low value until touchdown.

An unpowered aircraft in nearly level flight will decelerate at a rate given by:

$$\dot{V} = -\frac{g}{L/D} \qquad (1)$$

Thus, a vehicle with an L/D of 4.0 will decelerate at about 8 ft/sec^2, or about 5 knots per second. Dynamic pressure is proportional to velocity squared, and thus is rapidly decreasing. So, to maintain lift equal to weight, the angle of attack must be increased continuously. Furthermore, the low-aspect ratio of the HL-20 requires a greater increase in angle of attack than a conventional wing to effect a given change in lift. Also, the elevator trim angle will change significantly, since the touchdown speed is approximately 100 knots less than the approach speed, due to deceleration during the flare.

The result is that the landing maneuver in this vehicle requires a constantly increasing non-linear pitch rate.

The original baseline control system for this simulation consisted of a pitch rate feedback control law. This essentially generated an elevator command proportional to the difference between the pilot's stick input and the pitch rate of the aircraft. Although adequate for in-flight maneuvering, this system was somewhat difficult to land because it required ever increasing back-stick inputs during the landing flare.

An attempt was made to improve the system by adding an 'auto trim' feature. In this attempt, the existing elevator position was fed back through a first order filter and added to the existing elevator command. This resulted in a control law that would pitch at a rate proportional to stick deflection and hold a constant pitch attitude when the stick was released. This system was dubbed a 'Rate Command Attitude Hold' or RCAH control law.

Most aircraft are controlled and landed by reference to the pitch attitude. Pitch attitude is the parameter that can be observed most easily by the pilot. However, in the HL-20 pitch attitude is never constant during the landing flare, but must be increased at a non-constant rate and more rapidly at lower speeds. The RCAH system was unsatisfactory for landings and tended to induce oscillations.

The major aerodynamic forces during landing (lift and drag) are generated by the angle of attack. Although it cannot be directly observed by the pilot, angle of attack is an excellent control parameter for some flight modes. However, angle of attack must also be increased continuously at a non-constant rate in landing the HL-20.

The parameter that is actually being controlled in a landing is the flight path angle, or 'gamma', equivalent to the vertical direction of the velocity vector. The required gamma at any point in the landing is easily estimated. It is a constant value on the outer glide slope, and is roughly proportional to the altitude during the flare, reducing to zero at touchdown. If an aircraft can be designed to hold a given 'gamma command' it can be landed safely.

A pure 'gamma command' system is difficult to implement, however. A direct measurement of gamma itself is not easy to produce, and is sensitive to instrumentation errors and failures. It is also difficult to generate gamma commands with a control stick, as pilots generally expect a rate response from their inputs. It was found in the simulator

that some pilots tended to get out of phase during rapid maneuvers with the gamma command system, resulting in a pitch oscillation.

A better approach than pure 'gamma command' is to feed back the rate of change of gamma, or 'gamma dot'. Such a system is more natural to pilots, and maintaining gamma dot equal to zero is equivalent to holding gamma at a constant value.

The most direct digital method for generating gamma dot would be to numerically differentiate gamma itself. This method is even more sensitive than 'gamma command' to numerical problems, instrumentation errors, wind gusts, and pilot induced oscillations.

Some of these difficulties can be alleviated by passing gamma through a washout filter. Mathematically, this is equivalent to calculating the derivative of gamma and then feeding it through a first-order lag. This method, dubbed the 'Gamma Washout' law, was the first experimental law evaluated in the HL-20 study. It resulted in the easiest and best landings by most pilots (see Simulator Results).

However, the Gamma Washout system still requires an accurate measurement of gamma itself. A second method was developed which estimates gamma dot without requiring any knowledge of gamma. In nearly level flight, the rate of change of gamma can be estimated from the following relationship:

$$\dot{\gamma} = 57.3 \frac{g(N_Z - 1)}{V_{rel}} \qquad (2)$$

N_Z and V_{rel} are standard parameters that can be derived accurately from a number of possible sources. The combination of gamma dot, derived from N_Z, and pitch rate feedback, compared to the pilot's stick inputs, resulted in the 'NZQ' control law. This control law will pitch at a rate proportional to the pilot's stick deflections, and maintain one-G when the stick is released. Its performance is nearly the same as the Gamma Washout law. The only noticeable difference is that slight back pressure is required to maintain a constant sink rate as the ground is approached. This results in the NZQ law being less likely to produce 'ballooning' or pilot induced oscillations near touchdown.

It may be possible to improve the NZQ control law by additional filters, shapers, and turn coordination terms, but it appears to be acceptable in its simplest form.

Experimental Control Laws - Description

As shown in figure 8, the pilot's pitch stick input (δC) is first passed through a shaping function. The shaping function can be varied from a straight linear function to a pure quadratic ('square law') function, or anything between, depending on the 'square law ratio' (0 for a linear law, 1 for a quadratic law). A value of 0.6 was determined experimentally to have the most preferred characteristics.

The shaped column command is passed through a gain (GDQDC) to produce a pitch rate command. With the RCAH option there are no other active commands. In the Gamma Washout option, gamma is passed through a washout filter with a 1.0 sec time constant to produce an additional pitch rate command proportional to gamma dot. Likewise, in the NZQ option, gamma dot is estimated from N_Z and V_{rel} inputs (per equation above) and added to the pitch rate command signal. In either case a fixed gain, GDEGC, is used to control the amount of pitch rate required to maintain gamma dot approximately zero with no column inputs.

Figure 8. - Experimental Pitch Control Laws

If Autoland mode is engaged, a pitch rate commanded by the guidance program is added to the command signal. The pilot may make additional inputs during Autoland mode, without disengaging it.

The command signal is multiplied by a variable gain inversely proportional to dynamic pressure (Q-bar) to compensate for the variation in control effectiveness due to dynamic pressure. The gain (GQBAR1) is designed to produce a pitch rate step input response with a damping ratio of 0.7 and natural frequency of 3.14 rad/sec at 200 knots (the typical touchdown speed).

The elevator trim command ('auto trim') is determined by a lagged feedback of the present elevator deflection added to a speedbrake compensator. The speedbrake compensator consists of a gained washout of the speedbrake deflection, that drives the elevator trim whenever the speedbrake is in motion. The elevator feedback value is held constant at the value existing when 'weight-on-wheels' (WOW) is detected at main gear touchdown.

The total elevator command is the sum of the command input and the trim command. The elevator command is routed to the elevon actuators, which are simulated as first-order actuators with a time constant of 0.1 sec, rate limited to 200 degrees/second, and a deflection limit of 40° up or down.

Autoland Guidance & Control Laws

Figure 9 depicts the Autoland control laws used in the HL-20 simulation. These laws generate pitch rate and aileron deflection commands which are summed with other quantities in the inner loop pitch and roll stabilization programs.

Autoland was typically engaged and flown for the entire approach. The implementation allowed the pilot to make corrective inputs that were added to the autoland signal; this was not necessary as a rule, since the autoland, having a perfect "nav state" (i.e. navigation sensor and filter errors were not modeled), worked well. Disengagements of autoland and reversion to manual control were not attempted after an approach began.

The outputs of these autoland guidance laws are also used to drive a flight director on the EADI and on the HUD for the pilot's use, even when autoland was not engaged. The flight director depicts the difference between the state of the vehicle (flight path angle and bank angle) and the state commanded by the Autoland algorithm.

Vertical Steering

The pitch control law uses flight path angle as its primary reference. Gamma command is a good parameter to use for pitch guidance, since it is constant for most of the approach and is nearly zero at touchdown. By comparison, angle of attack or pitch commands must be varied as a function of airspeed and control deflections. Gamma command is also directly compatible with the Gamma Washout and NZQ control laws used for manual control in this simulation, and with the HUD velocity vector display which was used for out-the-window cues.

Figure 9. - Autoland Control Laws

Glideslope guidance consists of three segments--an outer glideslope with a fixed flight path angle of -17.0°, a 1.25 g pullup, and an inner glideslope of -1.0°. As shown in figure 9, the altitude of the aircraft is compared to the nominal glideslope altitude at the present downrange distance to produce an altitude error signal. The altitude error is multiplied by a fixed gain, and added to the nominal glideslope gamma to produce a preliminary gamma command.

The gamma command is then limited to avoid values that are either too steep (resulting in excessive airspeed and sink rate) or too shallow (resulting in a stall). The upper gamma command limit corresponds to flying at the maximum L/D condition until flare altitude is reached. Between flare and touchdown the upper limit is level flight, or zero gamma.

The gamma command lower limit is further limited by a 'floor' value proportional to altitude. If the aircraft is sinking at a rate that would impact the ground in a few seconds, the gamma command is reduced to initiate a pullup. At zero wheel altitude the floor value is equal to the desired gamma at touchdown, thus providing the final flare command.

Gamma rate is estimated from the vertical acceleration (N_Z) and earth relative velocity. To smooth performance, gamma is projected 0.5 sec ahead, by adding the present gamma plus 0.5 times gamma rate. The final pitch command sent to the inner loop is the difference between the gamma command and the projected gamma. A fixed gain in the flight control inner loop results in a pitch rate command, summed with other control inputs. The pilot may make inputs which are added to the Autoland command without disengaging Autoland.

After weight-on-wheels the Autoland law commands a 'slapdown' at 5 degrees/second, and after weight-on-nosegear (WONG) the Autoland commands are zeroed.

Lateral Steering

The lateral steering algorithm calculates aileron deflection commands to achieve a roll angle that maintains a ground track angle along the extended runway centerline (or 'localizer').

363

Figure 10. - Control Surface Mixing Laws

As shown in figure 9, the lateral displacement from centerline, Y_{RWY}, is multiplied by a gain to produce an centerline intercept angle command. This gain linearly increases below 5000 ft altitude to avoid unnecessary abrupt commands at higher altitudes, and still stay on the centerline as the runway is approached. The track angle command is the sum of the runway centerline heading, ψ_{RWY}, and the intercept angle command.

A track error is obtained by subtracting the actual ground track angle, ψ_{TRK}, from the commanded ground track angle. The resulting track error is resolved into ±180° and multiplied by a fixed gain to determine a roll angle command, which is limited to 40°.

The guidance aileron command is then obtained by multiplying the difference between the actual and commanded roll angles by a fixed gain. The aileron command is limited to 15° and sent to the roll flight control program, where it is summed with pilot inputs and roll stabilization commands. As in the longitudinal case, the pilot may make inputs which are added to the Autoland commands without disengaging Autoland.

Control surface mixer

The HL-20 conceptual design includes seven aerodynamic control surfaces, as shown in figure 1. These include an all-movable vertical fin, two wing flaps, and four body flaps - two located on the upper surface on the body and two

located on the lower surface. These body flaps are only capable of deflections outward, away from the body. This control surface arrangement is similar to earlier lifting bodies (ref. 2,3,4).

The control laws described above yield commands for an "aileron", "elevator", "speedbrake", and "rudder". Since the HL-20 has an unconventional control arrangement, these commands are passed through a controls mixer that separates these four commands into seven separate control surface commands.

The same control mixing arrangement was used for all subsonic control system configurations and is shown in figure 10. The wing flaps were used symmetrically for pitch control, and, at higher angles of attack, the upper body flaps were used for additional pitch control. The body flaps were deployed simultaneously for speedbrake and asymmetrically for roll control. The rudder was used for yaw control.

The use of wing flaps for roll control was investigated, but they showed a significant amount of adverse yawing moment. While the vehicle would initially roll in the commanded direction, this adverse yawing moment would result in a yaw in the opposite direction. Since the HL-20 has a strong aerodynamic "dihedral" effect, the resulting sideslip caused the vehicle to eventually roll in the opposite direction, against the commanded roll. Using asymmetric

body flaps for roll control showed very little adverse yaw, and was therefore implemented in this control law.

The body-flaps-as-speedbrake function is designed to provide an aerodynamic reaction force oriented in the longitudinal body axis with no pitch coupling. To achieve this goal, the upper and lower body flaps and a small amount of elevator (wing flaps) are deployed in a non-linear fashion in response to a speedbrake command. Lower body flaps are deployed proportional to the speedbrake command; upper body flaps are deployed proportional to the product of speedbrake command and a non-linear function of angle of attack. Speedbrake pitch trim compensation, using elevator deflection, is provided in the various flight control laws either explicitly or by use of a washout filter.

Examples of how the control surfaces respond to various control commands are shown in figure 11.
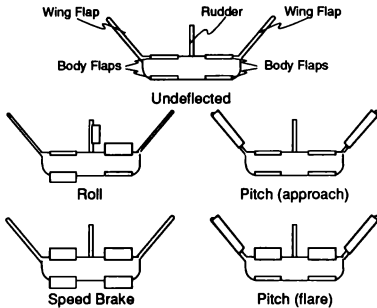
Figure 11. - Control Surface Mixing Strategy

Actuator Model

The same actuator model was used for all actuators in this study, and consisted of a 10 Hz first-order lag filter rate-limited to ±200 degrees/second. This high-performance actuator was chosen as an initial configuration, with a survey of the effects of lower actuator performance deferred until a later study. To accommodate this bandwidth, the actuator models were calculated twice in each major frame, for an effective minor frame rate of 66.7 Hz.

Simulator results

Using the initial set of manual control laws, several weeks of simulation sessions identified two design problems that needed attention. First, the original location of the landing gear resulted in significant nose wheel slapdown after landing (which the Shuttle crews call derotation) due to insufficient aerodynamic pitch authority. To alleviate this problem, the main gear location was shifted forward 1.5 feet. This correction greatly lessened the derotation problem.

The second design deficiency highlighted in the initial simulator study was that the original HL-20 design had a maximum lift-to-drag ratio of 3.2. This mandated relatively high-speed, steep approaches (25° at 347 knots equivalent airspeed) and minimal opportunity to flare, because of the relatively short amount of time available in the final flare maneuver due to energy bleedoff. Landing gear aerodynamic effects, which caused a decrease in lift-to-drag

to about 2.8, exacerbated the energy bleedoff problem. This focused attention on the need for a decrease in vehicle drag and prompted a successful effort to improve the HL-20 lift-to-drag ratio. The resulting design has a maximum lift-to-drag ratio of 4.3, and approaches are conducted on glideslopes of –17° at an equivalent airspeed of 300 knots. These approaches are very similar to those flown in the Shuttle Orbiter.

The simulator studies confirmed the need for an improvement in the original control laws. The introduction of the Gamma Washout control law significantly reduced the main gear touchdown sink rates while keeping main gear touchdown position dispersion roughly the same as the initial control law, as shown in figures 12 and 13. Note that these data are for the original 3.2 L/D max configuration, and represent a variety of wind and turbulence conditions, including 20 knots of head and/or crosswind and ±6 knots of turbulence.

Results from tests of the autoland system compare well with manual landings (figures 14-17). These include winds up to 25 knots, from various directions, as well as ±12 knots of turbulence. These data were obtained at the higher 4.3 L/D configuration.

A parametric study was performed in which the L/D of the vehicle was artificially varied around the original 3.2 value (ref. 7). This showed a strong effect of lift-to-drag ratio upon pilot rating for this type of lifting body and predicted Level 1 Cooper-Harper ratings for configurations with L/D above 3.8 . This has been borne out in later piloted studies of the higher 4.3 L/D configuration, in which Cooper Harper ratings of 2 to 3 in the landing task are consistently given by a variety of pilot subjects.

Early tests of off-nominal energy conditions at the beginning of the landing approach indicate the HL-20 has ample energy margins to cope with unexpected winds and navigation errors. A recent test yielded a successful landing from an initial condition with a 29 percent energy deficiency (2,446 feet below glideslope at 10,000 feet, and an airspeed deficiency of 80 knots equivalent). The large drag-producing capability of the combined upper and lower body flaps shows promise for high-energy conditions as well.

Concluding Remarks

The real-time simulator has provided tangible evidence of the feasibility of unpowered horizontal landings in the HL-20 with relatively simple flight control laws. Additional improvements in the control laws have resulted in a vehicle with superior flying qualities under adverse conditions, including off-nominal energy and crosswind conditions. An "N7Q" control law, with pitch rate feedback for stability, seems to provide optimal flying qualities with minimal sensor requirements. With the addition of flight director symbology projected on a heads-up display, successful manual landings with little training could be performed with Level 1 flying qualities.

These early tests must be repeated when improved actuator and navigation sensor, and landing gear models are developed; however, basic vehicle landing capabilities (glide performance, energy margins, handling qualities, and visibility) appear to be more than adequate as a starting point for a next-generation spacecraft design.

Utilization of simulation tools in the conceptual design of the HL-20 vehicle has resulted in early identification of deficiencies and opportunities for improvement in the vehicle, avoiding the cost and impact of design changes if
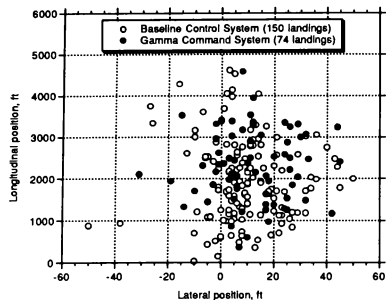
Figure 12. – Effect of improved longitudinal
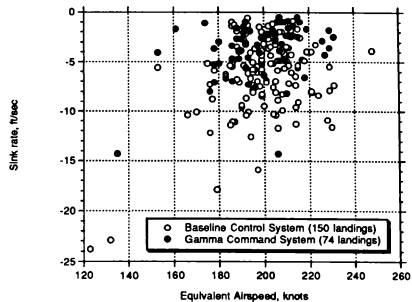control system on landing dispersion; L/D ~ 3.2

Figure 13. – Effect of improved longitudinal control system
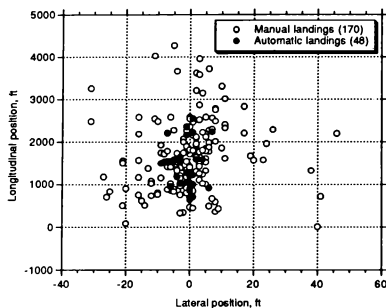upon touchdown velocities; L/D ~ 3.2

Figure 14. – Touchdown dispersion for manual and automatic landings
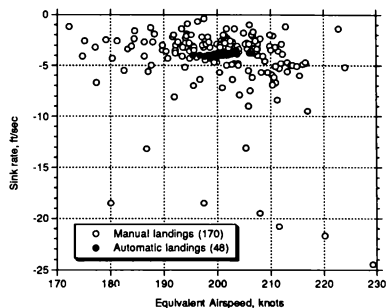in various wind and turbulence conditions; L/D ~ 4.3

Figure 15. – Touchdown velocities for manual and automatic landings
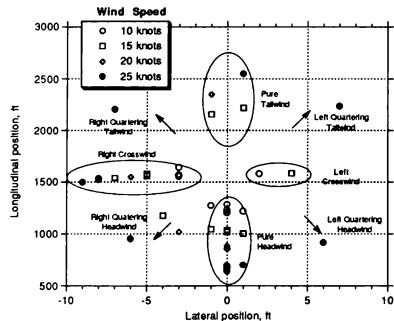in various wind and turbulence conditions; L/D ~ 4.3

Figure 16. – Autoland touchdown dispersion: 48 landings
moderate turbulence, various winds, L/D ~ 4.3
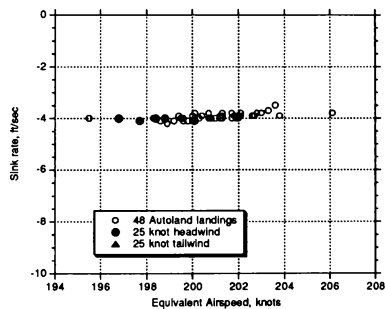
Figure 17. – Autoland touchdown velocities: 48 landings
moderate turbulence, various winds, L/D ~ 4.3

they had been made later in the program. In addition, use of a real-time simulator has demonstrated the satisfactory flying and landing characteristics of this type of lifting-body vehicle.

References

[1]Manke, John A.; Retelle, John P.; and Kempel, Robert W.: Assessment of Lifting Body Vehicle Handling Qualities. AIAA Paper No. 71-310, 1971.

[2]Kemple, Robert W.: Flight Evaluation of HL-10 Lifting Body Handling Qualities at Mach Numbers from 0.30 to 1.86. NASA TN D-7537, 1970.

[3]Nagy, Christopher J.; and Kirsten, Paul W.: Handling Qualities and Stability Derivatives of the X-24B Research Aircraft. AFFTC-TR-76-8, 1976.

[4]Smith, Harriet J.: Evaluation of the Lateral-Directional Stability and Control Characteristics of the Lightweight M2-F1 Lifting Body at Low Speeds. NASA TN D-3022, 1965.

[5]Powell, Richard W., Cruz, Christopher I.: Guidance and Control Analysis of the Entry of a Lifting Body Personnel Launch Vehicle, AIAA-91-0055, January 1991

[6]Jackson, E. Bruce and Cruz, Christopher I.: Preliminary Subsonic Aerodynamic Model for Simulation Studies of the HL-20 Lifting Body. NASA TM-4032, 1991

[7]Jackson, E. Bruce; Rivers, Robert A.; and Bailey, Melvin L.: Effect of Lift-to-Drag Ratio Upon Pilot Rating for a Preliminary Version of the HL-20 Lifting Body. AIAA-91-2890, August 1991

MODELING AN AUTOPILOT AND THRUST COMPENSATOR IN AN AUTOMATIC CARRIER LANDING SYSTEM

John L. Crassidis

D. Joseph Mook

Research Assistant
Member AIAA

Associate Professor
Member AIAA

Department of Mechanical and Aerospace Engineering
State University of New York at Buffalo, Buffalo, NY 14260

Abstract

A detailed computer simulation which includes the aircraft flight dynamics, pitch attitude autopilot, and automatic thrust compensator is presented for an F-4 aircraft under automatic carrier landing system control. The aircraft simulation is first derived by utilizing a full six degree-of-freedom rigid-body model. This model is then increased to include the pitch autopilot and automatic thrust compensator. The control variables for these systems are the elevator, which generally controls pitch angle, and the thrust, which generally controls angle of attack. A detailed digital computer simulation, verified with frequency domain techniques and test data, allows the replacement of simplified transfer function models for use in an automatic carrier landing simulation. Therefore, internal states and dynamics associated with the aircraft subsystems can be evaluated.

Introduction

The autopilot and thrust compensator are essential components in an automatic carrier landing system (ACLS). The ACLS is a fully automatic control system that provides flight path control until touchdown on an aircraft carrier (see, e.g., [1-2]). Primary ACLS components include a shipboard radar tracking system, a digital computer, and a radio data link. The radar is used to measure the aircraft's position. The digital computer then calculates corrective control commands, which are transmitted to the aircraft. The aircraft's autopilot and thrust compensator respond to these flight control commands, which directly control the aircraft's position and

orientation. The main goal of the ACLS is to automatically control the aircraft in adverse conditions such as pilot fatigue, low visibility weather, heavy atmospheric turbulence, and/or deck motion caused by high seas.

Previously, ACLS simulations [3] used a reduced model for the aircraft simulation. This model, obtained from experimental Bode data [5], incorporated a 3rd order transfer function relating the aircraft's altitude response to the transmitted pitch control command. However, the reduced model neglects the dynamic effects of the autopilot and thrust compensator. Of particular interest is the aircraft's response to atmospheric turbulence. With the transfer function model, used in [3], the aircraft's response to turbulence is solely controlled by the ACLS shipboard controller. But, in practice, the automatic thrust compensator also minimizes the aircraft's response to turbulence. The transfer function model neglects this *internal* control system. In addition, new tracking filter developments (see [6]) may significantly reduce noise sensitivity in aircraft performance and tracking accuracy. This flight dynamics-based tracking filter requires on-board aircraft sensor information (angle of attack and airspeed). In order to incorporate this tracking filter into the ACLS, a detailed simulation of the aircraft dynamics, autopilot and thrust compensator is required.

In this paper, the fundamental elements and feedback control systems of the pitch autopilot and thrust compensator are shown and then incorporated into a computer-generated simulation. The simulation is first presented for an open-loop study by a numerical integration scheme of the rigid body equations of motion (due to space limitations, the simulation is shown in the vertical altitude plane only, although the techniques are easily extended to the horizontal plane). The closed-loop system results are next shown, so that the simulation can be analyzed

to flight-verified data of an F-4 aircraft in the form of frequency response data.

Description of the Simulation

Aircraft Model

The aircraft simulation is derived by using a 12th order state space model, based on the six degree-of-freedom rigid-body aircraft dynamic equations of motion and the concept of static stability using aircraft forces and

moments (see [7] or [8] for more details). All computer simulation trajectories are produced with an F-4 aircraft, due to the availability of experimental ACLS information. Table 1 summarizes these aircraft equations of motion, where (α, β) are angle of attack and sideslip angle, respectively, (p, q, r) are angular velocities, (Φ, Θ, Ψ) are roll, pitch and yaw angles, $(\delta_e, \delta_a, \delta_r)$ are elevator, aileron and rudder angles, $(\bar{V}, \bar{T})$ are airspeed and thrust, and $(\ddot{x}, \ddot{y}, \ddot{z})$ are translational accelerations in an inertial reference frame. The symbols $(k_{i'_s}$ and $b_{i'_s})$ represent aircraft forces, moments and other constants (see [8] for more details).

Table 1 Aircraft Equations of Motion

$$\dot{p} = \left[-\left(I_{zz} - I_{yy}\right)qr\right]/I_{xx} + \left(k_1\beta + k_2p + k_3r + b_1\delta_a + b_2\delta_r\right)0.5\left(\bar{V}\right)^2$$

$$\dot{q} = \left[-\left(I_{xx} - I_{zz}\right)pr\right]I_{yy} + \left(k_4\alpha + k_5q + b_3\delta_e\right)0.5\left(\bar{V}\right)^2 + b_4\bar{T}$$

$$\dot{r} = \left[-\left(I_{yy} - I_{xx}\right)qr\right]I_{zz} + \left(k_6\beta + k_7p + k_8r + b_5\delta_a + b_6\delta_r\right)0.5\left(\bar{V}\right)^2$$

$$\dot{\Phi} = p + q\sin\Phi\tan\Theta + r\cos\Phi\tan\Theta$$

$$\dot{\Theta} = q\cos\Phi - r\sin\Theta$$

$$\dot{\Psi} = \left(q\sin\Phi + r\cos\Phi\right)/\cos\Theta \qquad (1)$$

$$\ddot{x} = \left[k_{10} + k_{11}\alpha + k_{12}\beta + b_7\delta_a + b_8\delta_r + b_9\delta_e\right]0.5\left(\bar{V}\right)^2 + b_{10}\bar{T}$$

$$\ddot{y} = \left[k_{13} + k_{14}\alpha + k_{15}\beta + b_{11}\delta_a + b_{12}\delta_r + b_{13}\delta_e\right]0.5\left(\bar{V}\right)^2 + b_{18}\bar{T} + g$$

$$\dot{x} = \ddot{x}$$

$$\dot{y} = \ddot{y}$$

$$\dot{z} = \ddot{z}$$

In order to evaluate the aircraft equations of motion for an actual aircraft trajectory, an open-loop simulation is first conducted utilizing F-4 aircraft coefficients. All trajectories are initially started with the aircraft set to trimmed values. For ACLS landing the final approach speed is approximately 220 ft/sec (the trim values for angle of attack and elevator setting are 12.7 and −10.3 degrees, respectively). A sample trajectory for a 1 degree step elevator input, relative to trim, is shown in Figure 1. The step (transient) response characteristics are similar to an actual F-4 aircraft undergoing similar maneuvers. Other computer simulated characteristics (not shown here), such as airspeed, angle of attack, etc., also reflect the actual aircraft responses (see [13] for more details).
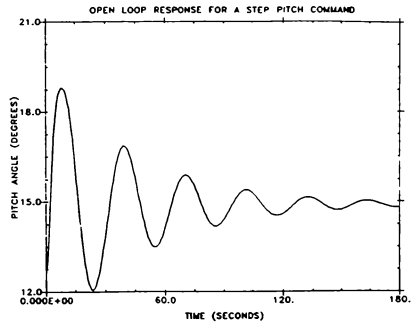

Figure 1 Open-Loop Pitch Angle Response

Pitch Autopilot

The basic block diagram for the closed-loop system of the pitch autopilot with attitude feedback is shown in Figure 2. The pitch autopilot operates as follows. A desired pitch command is first sent to the aircraft. A pitch angle error signal, representing the difference between the measured pitch angle and the desired pitch angle, is the input to the autopilot controller. The controller commands changes in the elevator setting, relative to a datum (typically, the trim value). The aircraft then responds to the new elevator setting with changes in pitch angle, pitch rate, vertical acceleration and angle of attack.

Closing the loop using only attitude feedback achieves the desired pitch angle. But, since the aircraft has very little natural damping (see Figure 1), the closed-loop response characteristic also has a low damping ratio. The dynamic performance of the aircraft can be severely degraded, even causing the system to become unstable (this is later shown by frequency domain techniques). To achieve significant damping and dynamic performance, a pitch rate feedback is also provided (represented by the inner loop of the block diagram in Figure 2).
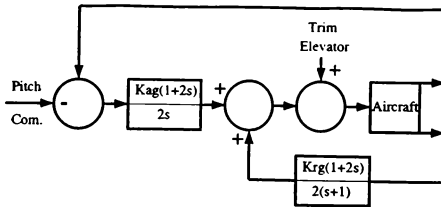
where K_{ag} is the attitude gain, K_{rg} is the pitch rate gain and Θ_c is the input pitch command.

The autopilot simulation is accomplished by converting the aircraft equations of motion (Table 1) into state space form and numerically integrating. The autopilot is engaged when the aircraft is trimmed in straight and level flight ($\Theta_0 = 12.7$ degrees). The input to the feedback system is a pitch change command relative to the trim level. The elevator command is also implemented as a relative change from the initial setting:

$$\delta_{e_{new}} = \delta_{e_{trim}} + \Delta \delta_e \qquad (3)$$

Simulation results are shown in a later section.

Frequency Domain Analysis

Frequency domain techniques are used in order to evaluate the response characteristics of the pitch attitude autopilot closed-loop system. The frequency domain characteristics of the aircraft are characterized by the short period and phugoid modes. The corresponding transfer functions are derived by linearizing the nonlinear aircraft equations of motion (see [7]). Therefore, local stability tests are possible by replacing the nonlinear aircraft model with the linear approximations.

The feedback loops in the autopilot tend to destabilize the response of the aircraft in the short period mode [11]; therefore, the aircraft response is analyzed by utilizing the short period approximation, derived as [8]:

$$\frac{\Theta(s)}{\delta_E(s)} = -\frac{2.944s + 1.00876}{s(s^2 + 2.832 + 8.702)} \qquad (4)$$

with $\omega_{n_{sp}} = 2.95$ rad/sec $\quad \zeta_{sp} = .48$

The negative sign is due to the fact that a negative change in elevator setting causes a positive change in pitch angle.

After extensive research, a relationship has been developed [7] relating the damping ratio and natural frequency to flying quality as illustrated in Figure 3. The intersection of the natural frequency and damping ratio for the short period mode in Equation (4) yields a good flying characteristic for this aircraft.
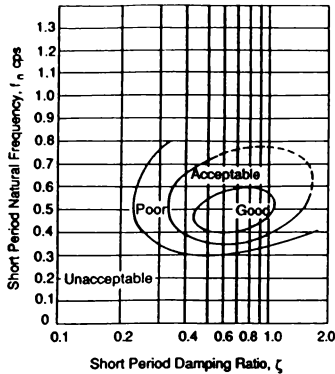

Figure 2 Block Diagram of the Pitch Autopilot with Rate Feedback

The elevator controller equation in the frequency domain is given by [5]:

$$\Delta \delta_e = \left[K_{ag}(\Theta - \Theta_c) + \frac{sK_{rg}}{s+1}\dot{\Theta} \right] \left[\frac{1+2s}{2s} \right] \qquad (2)$$

Figure 3 Short Period Flying Quality (from Nelson [7])

The autopilot closed-loop response in the the frequency domain is analyzed by utilizing the block diagram illustrated in Figure 2. The feedback loop for the pitch angle response is:

$$\frac{\Theta}{\Theta_c} = \frac{(I.A.)(A.F.)}{1 + K_{ag}(I.A.)(A.F.)} \qquad (5)$$

where $A.F.$ is the aircraft model in the frequency domain (R.H.S. of Equation 4) and $I.A.$ is the aircraft actuator, given as [5]:

$$I.A. = \frac{\delta_E(s)}{v(s)} = \frac{1}{\tau_h s + 1} \qquad (6)$$

where the input of this transfer function is an amplified voltage representing the elevator servo and the output is the change in elevator setting (in degrees). The servo time constant (τ_h) for an F-4 is approximately 0.1 seconds.

The attitude gain (K_{ag}) is determined by using a root locus plot of the closed-loop transfer function. Figure 4 depicts the root locus plot for the short period mode. As the attitude feedback gain increases, one pole of the transfer function becomes more stable into the left half of the complex plane. However, the system stability for the remaining poles decreases rapidly and eventually becomes unstable. This analysis proves the need for a pitch rate feedback loop to improve system stability. The attitude gain results in a stable system response between a range of 0 to 5 deg/deg. The optimal gain occurs when

all three poles align perpendicular to the real axis. This occurs at a value of approximately 3.25 deg/deg. The actual gain used in the F-4 is 3.5 deg/deg [5].
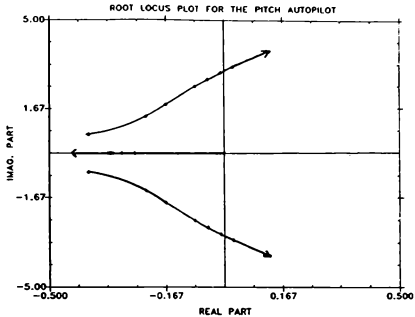

Figure 4 Root Locus Plot for the Pitch Autopilot

Experimental data points in terms of a Bode plot for the longitudinal pitch autopilot are provided in [5]. This is compared to the computer simulation Bode data (accomplished by sinusoidal excitations at various magnitudes and frequencies). The resulting magnitude and phase plots are illustrated in Figure 5 and 6 (solid line is the simulated solution and dashed is the experimental data). The frequency response characteristics of the simulated pitch autopilot closely match the measured test data.
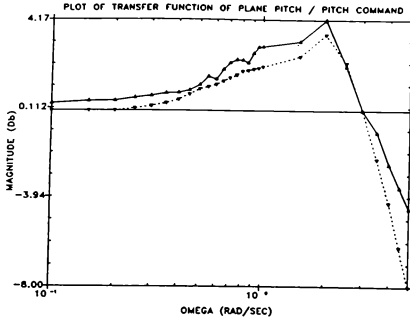
PLOT OF TRANSFER FUNCTION OF PLANE PITCH / PITCH COMMAND

**Figure 5 Magnitude Plot for
Pitch Angle to Pitch Command**

PLOT OF TRANSFER FUNCTION OF PLANE PITCH / PITCH COMMAND
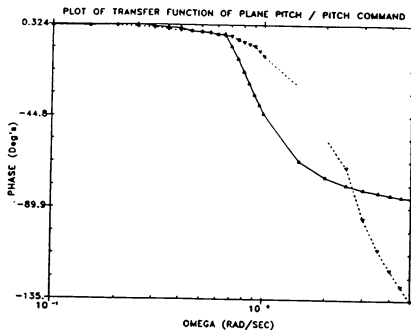
**Figure 6 Phase Plot for Pitch
Angle to Pitch Command**

Thrust Compensator

The autopilot achieves the desired pitch angle, but the inertial flight path angle (γ) must be controlled in order to land the aircraft using the ACLS. One method of achieving a desired flight path angle is to control the angle of attack to a desired reference point (e.g., the trim value) while also controlling pitch angle, since the flight path angle represents the difference between the pitch angle and angle of attack. Angle of attack is controlled by utilizing the thrust compensator.

Controlling the angle of attack is also desirable in the case of atmospheric disturbances such as wind gusts, turbulence, and the extreme conditions of wind shear. These disturbances may occur during ACLS operation since a carrier landing system must be able to function in all adverse weather conditions. The variations in wind speed and direction are measured in the vertical and horizontal directions [9]. This phenomenon has been a major contributor to several commercial and military aircraft crashes. In a wind shear an aircraft can severely lose lift, due to dramatic fluctuations in airspeed. The lift of an aircraft is also a function of the angle of attack. Therefore, the lift can be regained by varying the angle of attack.

The thrust compensator commands thrust changes according to the control law [5]:

$$\Delta T_c = \left[\frac{A1}{s} + \frac{A2}{s+1} + \frac{A3}{T3s+1}\right]\Delta\alpha - \frac{Kn}{1.2s+1}N_s$$
$$+ \frac{K}{.25s+1}\Delta\delta_e \tag{7}$$

where $\Delta\alpha$ is the detected angle of attack error (measured vs. commanded), N_s is the vertical acceleration of the aircraft, $\Delta\delta_e$ is the change in elevator setting commanded by the pitch autopilot, and K, Kn and $A1 - A3$ are feedback gains. In practice, N_s is measured by a vertical accelerometer expressed per unit of gravity and is used to sense changes due to turbulence. A positive change in acceleration causes a negative change in the commanded thrust. The elevator feedback loop's (shown by the last term in Equation 7) role in the coupling between the autopilot and the thrust compensator is explained in a later section.

The compensator gains and time constants vary for changing angle of attack error. These values are shown in Table 2.

**Table 2 Thrust Compensator
Gains and Time Constants**

	A1	A2	A3	Kn	K	T3
alpha below ref.	337	2443	242	46533	776	0.72
alpha above ref. and inc.	337	2443	1945	46533	776	0.29
alpha above ref. and dec.	337	2443	1945	46533	776	0.72

The controller, coupled with the pitch rate feedback loop, commands changes in the elevator setting. This command is added to the trim elevator setting. The aircraft then responds to the new elevator setting, producing changes in pitch angle, pitch rate, vertical acceleration and angle of attack. This part of the closed-loop is represented by the pitch displacement autopilot with the feedback loop given by Equation (2).

The automatic thrust compensator does not respond to a pitch command, but rather to a change in angle of attack or vertical acceleration. This can be delayed for an indefinite period, based upon the aircraft's response time. In order to create a faster response time in the thrust compensator, an elevator feedback loop is used. This feedback loop leads the aircraft's response, so that the automatic thrust compensator is engaged *before* a change in angle of attack or a change in vertical acceleration occurs. These thrust commands are then summed and converted by a transfer function to an actual thrust change in the simulation. The loop is then closed by feeding back this thrust change to the aircraft model. The autopilot and automatic thrust compensator closed-loop operation continues until the desired angle of attack is achieved.

The first row of Table 2 is for an angle of attack error that is negative. The second row is for a positive angle of attack error that is decreasing. The third row is for a positive angle of attack error that is increasing.

To model the relationship between the thrust command input to an actual thrust change, a second order engine time lag transfer function is utilized [5]:

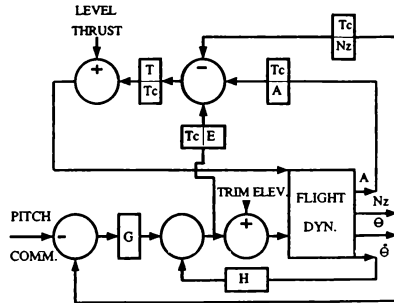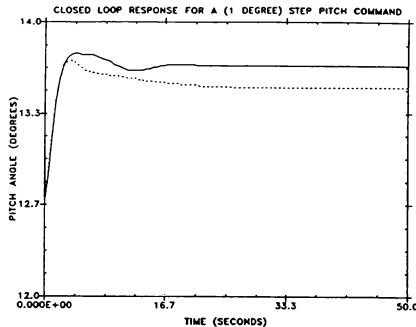$$\frac{\Delta T}{\Delta T_c} = \frac{1}{.044 s^2 + .0973 s + 1} \qquad (8)$$

The steady state value for this transfer function equals the magnitude of the commanded thrust input.

Coupling of the Autopilot and Thrust Compensator

The combined closed-loop system incorporates the pitch autopilot with pitch rate feedback and the thrust compensator. The block diagram for this system is shown in Figure 7. A pitch command is first sent to the pitch autopilot. A pitch angle error signal, representing the difference between the measured pitch angle and the commanded pitch angle, is the input to the controller.
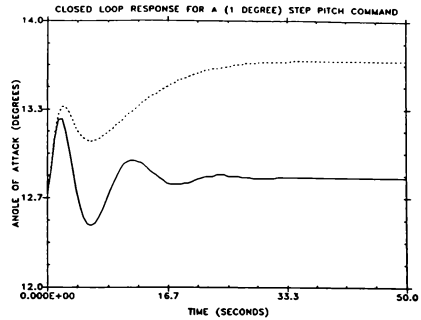

Figure 7 Block Diagram of the Combined Closed-Loop System

Simulation Results

The combined closed-loop response for a 1 degree step pitch command is shown in Figure 8 (the dashed line is the response utilizing the autopilot exclusively, and the solid line is the response of the combined autopilot/thrust compensator). By incorporating the automatic thrust compensator, the steady state pitch error is near zero. The angle of attack response for this step input is shown in Figure 9. With the autopilot only (dashed line), the combined angle of attack and pitch angle does not enable the control of the inertial flight path angle. The thrust compensator controlled the angle of attack back to the desired trim value (the compensator can control the angle of attack to any desired setting). This combined closed-loop now enables the control of the inertial flight path angle to any desired value. Therefore, the autopilot/thrust compensator simulation can be expanded to simulate the carrier landing process (shown in [10]).

Figure 9 Combined Closed-Loop
Angle of Attack Response

Though the aircraft model is a set of nonlinear differential equations, the response characteristics of the pitch autopilot are linear to the input command signal. This feature is illustrated, using different magnitudes for the step pitch command, in Table 3.

**Table 3 Various Characteristic
Responses for Different Step Inputs**

Step Pitch Com.	Rise Time (Sec.)	Settling Time (Sec.)	Percent Over- Shoot	Percent Steady Error
1 Deg.	3.6	18.5	23%	15%
4 Deg.	3.5	18.5	25%	18%
-3 Deg.	3.6	18.0	23%	15%

The vertical acceleration response for a step pitch command is shown in Figure 10. The initial drop in vertical acceleration is caused by a pitching moment induced when changing the elevator setting (i.e., a positive pitching moment causes an initial drop in vertical acceleration). The response without the automatic thrust compensator (dashed line) is lightly damped ($\zeta \approx 0.2$). But, with the addition of the thrust compensator (solid line) the settling time for the system is faster due to a higher damping ratio ($\zeta \approx 0.7$). This improved acceleration characteristic also provides an increase in stability

Figure 8 Combined Closed-Loop
Pitch Angle Step Response

in the aircraft response to atmospheric disturbances, such as turbulence. The thrust history, from the combined closed-loop response, for a step pitch command is shown in Figure 11. Since the response of the angle of attack initially increases and is also coupled with an initial decrease in vertical acceleration, the thrust increases. The steady state value for the thrust is 8632 lbs. Therefore, the thrust increased a net amount of 552 lbs from the initial level flight thrust setting (8080 lbs). This increase in thrust causes the aircraft to gain altitude, shown in Figure 12. The magnitude of the airspeed response, shown in Figure 13, is not significantly changed. This is ideal for an automatic landing on a carrier.

Atmospheric disturbances such as wind gusts and turbulence are modeled by expressing the velocity components in terms of inertial and gust velocities (see [12] for more details). These wind gusts can be represented as a sharp edge gust with a linear decay ($M_g = -At + B$) or as a continuous sinusoidal gust ($M_g = A \sin \omega_g t$). The modeled wind gusts simulate the aircraft response to a disturbance input.

The step pitch command response in addition to a 10 ft downward wind gust (from 0 to 5 seconds with a 1 ft/sec decay) are shown in Figure 14 and Figure 15. The angle of attack initially has an offset error of about −2 degrees, induced by the downward gust. Again, the thrust compensator controls the angle of attack (Figure 15) to the desired trim value. Other disturbance responses (not shown here) indicate that the aircraft simulation exemplify the actual aircraft response under similar disturbance inputs [13].
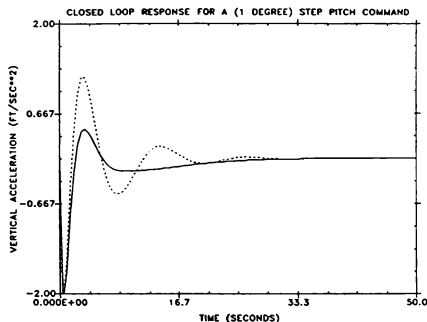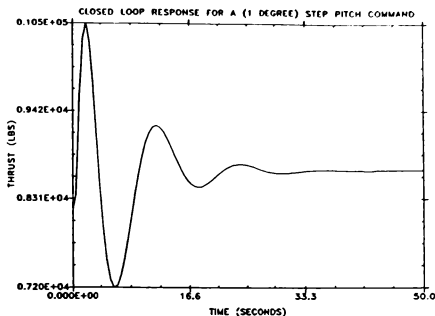

Figure 11 Closed-Loop Thrust History

**Figure 10 Closed-Loop
Vertical Acceleration Response**
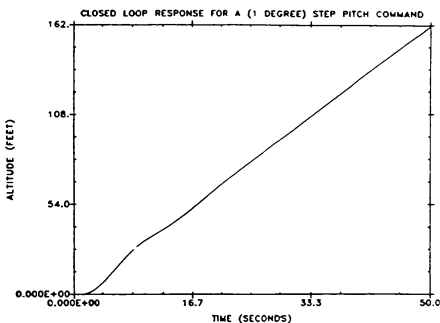
Figure 12 Closed-Loop Aircraft Altitude Response

CLOSED LOOP RESPONSE FOR A (1 DEGREE) STEP PITCH COMMAND
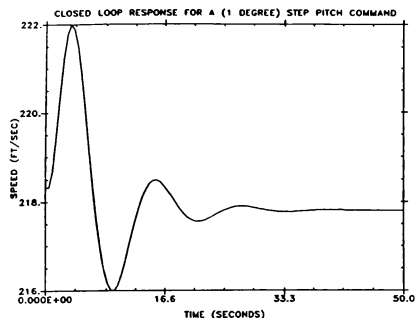
Figure 13 Closed-Loop Aircraft Airspeed Response

CLOSED LOOP RESPONSE FOR A (1 DEGREE) STEP PITCH COMMAND

Figure 15 Closed-Loop Angle of
Attack Response with a Wind Gust

CLOSED LOOP RESPONSE FOR A (1 DEGREE) STEP PITCH COMMAND
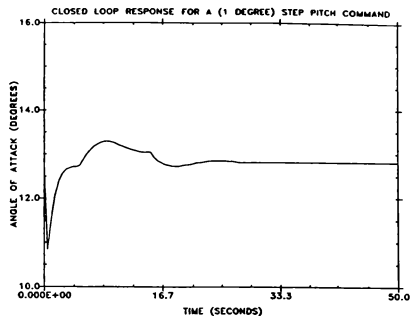
Figure 14 Closed-Loop Pitch
Response with a Wind Gust

Conclusions

In this paper, a detailed simulation of an F-4 aircraft with a pitch autopilot and thrust compensator has been presented. This system is first developed with an open-loop simulation of an aircraft using a 12th order state-space model. Then, a closed-loop simulation is developed for the aircraft under autopilot/thrust compensator control. The autopilot maintains a desired pitch attitude, while the automatic thrust compensator maintains the desired angle of attack and minimizes vertical acceleration changes in the aircraft. The combination of these systems is essential for an automatic flight-path control landing. Available frequency domain test data indicate that the simulation is accurate for an actual aircraft response undergoing similar maneuvers. Therefore, the aircraft simulation can be incorporated in an automatic carrier landing system simulation in order to investigate aircraft tracking and control performance.

References

[1] Urnes, J.M. and Hess, R.K., "Development of the F/A-18A Automatic Carrier Landing System," **AIAA Journal of Guidance, Control, and Dynam-ics, Vol. 8, No. 3, pp. 289–295, May-June 1985.

[2] Urnes, J.M., Hess, R.K., Moomaw, R.F., and Huff, R.W., "H-Dot Automatic Carrier Landing System for Approach Control in Turbulence," **AIAA Journal of Guidance, Control, and Dynamics**, Vol. 4, No. 2, pp. 177–183, March-April 1981.

[3] Mook, D.J., Swanson, D.A., and Roemer, M.J., "Improved Noise Rejection in Automatic Carrier Landing Systems," to appear in the **AIAA Journal of Guidance, Control, and Dynamics.**

[4] Mook, D.J., Shyu, I.M., and Coye, K.B., "A Nonlinear Aircraft Tracking Filter Utilizing Control Variable Estimation," to appear in the **AIAA Journal of Guidance, Control, and Dynamics.**

[5] "Automatic Carrier Landing Compatibility," **Bell Aerospace**, Buffalo, N.Y., Final Report #60003–180, Feb. 1969.

[6] Mook, D.J., and McGrath, J., "Vertical State Filters for ACLS Utilizing Airplane Force Models and On-Board Sensors," Final Report, Bell-University at Buffalo Research Agreement, 9103–246, Jan. 1990.

[7] Nelson, R.C., **Flight Stability and Automatic Control**, McGraw, 1987.

[8] Roskam, J., **Airplane Flight Dynamics and Automatic Control, Part 1 and 2**, Roskam Aviation and Engineering Corp., The University of Kansas, Lawrence, 1979.

[9] "Automatic Carrier Landing System Computer Program Improvements," **Bell Aerospace**, Buffalo, N.Y., Final Report #6500–927243, Feb. 1982.

[10] Crassidis, J.L., and Mook, D.J., " An Automatic Carrier Landing System Utilizing Aircraft Sensors", to appear, Proceedings of the **AIAA Flight Simulation Technologies Conference**, New Orleans, 1991.

[11] Loeb, J.L., "Automatic Landing Systems are Here," **AGARD Conference Proceedings on Aircraft Landing Systems**, No. 59, 1969.

[12] Sheftel, D.J., "New Guidance Development for All Weather Landing," **AGARD Conference Proceedings on Aircraft Landing Systems**, No. 59, 1969.

[13] Crassidis, J.L., "Tracking and Control in an Automatic Carrier Landing System Utilizing Aircraft Sensor Information," Master's Thesis, Department of Mechanical and Aerospace Engineering, State University of New York at Buffalo, Buffalo, N.Y., Jan. 1991.

A COMPARISON OF SEVERAL NUMERICAL INTEGRATION ALGORITHMS EMPLOYED IN REAL TIME FLIGHT SIMULATION: ESPECIALLY INCLUDING THEIR IMPACT ON EFFECTIVE DELAY AND SIMULATION ACCURACY

Frank M. Cardullo*
Mark Kaczmarek**
State University of New York
Binghamton, New York

Brian J. Woycechowsky
Binghamton, New York

Abstract

This paper reports on the results of a study which examined the performance of several integration algorithms, both old and new, and compares their performance using both functions which are analytically integrable and aircraft accelerations developed from a full force and moment model of a high performance aircraft. The comparison against the analytic function permits an assessment of the various discrete approximations to a continuous solution. The algorithms investigated include;
  a) second order Adams-Bashforth
  b) advanced Euler
  c) sinusoidal
  d) Howe modified Euler

It was found that there was little difference in the performance of the various methods, at small Δt, other than Advanced Euler and Howe Modified Euler provided some lead which is advantageous in minimizing cue delay.

Introduction

The criteria against which integration algorithms are evaluated are accuracy, stability, trans-port delay and phase shift. In the past the selection of integration algorithms for real time simulation was motivated by a desire to achieve the best accuracy possible at the lowest update rate possible. With the advances in computer technology of the last decade, much higher update rates are achievable at a reasonable cost. In addition, higher update rates are required to minimize the transport delay because of its effects on pilot performance in manned flight simulators. One result of higher update rates is that the selection of an integration algorithm on the basis of accuracy is less demanding.

It will be illustrated that at update rates in the vicinity of 60Hz the error is quite small and undetectable by the pilot. Since in piloted vehicles the operation is constantly maneuvering, the accuracy issue is relatively minor. This statement is not valid for spacecraft simulation when integration continues for long periods of time along a specific trajectory. Other applications where accuracy is important are simulation of pilot launched guided and unguided munitions. The accuracy of this latter category is more demanding than for integrating a piloted aircraft state vector because of scoring requirements. Position accuracy for

*Associate Professor, Mechanical Engineering, Associate Fellow AIAA
**Graduate Student, Sr Member AIAA

space flight mechanics is also very important because of orbit closure and precision maneuvers such as rendezvous and docking.

At these high update rates, stability of most integration algorithms is also not a major problem. Hence, the major thrust of choosing an integration algorithm is the minimization of transport delay and phase shift.

There are probably as many numerical integration algorithms as there are people who have carefully analyzed the problem.

The paper compares the performance of five different integration methods, all of which are either commonly used in, or specifically developed for real time, man in the loop, simulation: second order Adams-Bashforth, advanced Euler, sinusoidal predictor, sinusoidal corrector and Howe modified Euler. The comparison will be performed in two scenarios, first in simulating the response of a lightly damped, second order system and second in simulating the response of a high performance aircraft to a step input in elevator. The natural frequency and damping ratio of the damped sinusoidal response was chosen to closely approximate the aircraft's longitudinal mode response. The aircraft simulation involves a six degree of freedom, full force and moment, non-linear formulation yielding aircraft accelerations in the body axes which are subsequently integrated twice.

Discussion

The damped sinusoidal response was implemented employing the classical second order non-homogeneous differential equations:

$$\ddot{x} + 2\zeta\omega_n\dot{x} + \omega_n x = \omega_n^2 y \quad (1)$$

Equation (1) was mechanized for solution by first solving for the highest order derivative and then integrating to obtain $\dot{x}$ then once again to obtain x.

$$(2)$$
$$\ddot{x} = \omega_n y - 2\zeta\omega_n\dot{x} - \omega_n^2 x$$

$$\dot{x} = \int_{t_1}^{t_2} \ddot{x}dt \quad (3)$$

$$x = \int_{t_1}^{t_2} \dot{x}dt \quad (4)$$

The values of the parameters of equation (2) are; ω_n = 2.2 rad/sec, ζ = 0.2 and y = 0.45. The integrations indicated by equations (3) and (4) are performed using the various numerical integration techniques listed above. These results are compared against a "continuous" solution provided by employing the step response algorithm of PC-MATLAB.

The equations for the five algorithms are presented here with some discussion. If more information is required regarding the derivation of the various algorithms, the references may be consulted. The advanced Euler algorithm is so titled because it advances the use of the results of the integration one frame for the first integration and two frames for the second integration with respect to the classical Euler method. The classical Euler method has the form:

$$x_{n+1} = x_n + \Delta t \dot{x}_n \quad (5)$$

and the advanced Euler algorithm consists of the equation;

$$x_n = x_{n-1} + \Delta t \dot{x}_n \quad (6)$$

In equation (5) the integrated value of the derivative ($\dot{x}$) is

time tagged to be used in the frame subsequent to the one in which it is computed. The formulation of equation (6) is essentially the same as equation (5) except that the integrated value of the derivative is time tagged to be used in the current frame. Hence, the result is advanced one frame. If two sequential integrations are performed in this fashion, then the value of the second integration is advanced two frames. This is especially useful to provide advanced displacement information of the visual system image generator since typically these devices have throughput delays of about three display fields. Therefore, it can be concluded from the foregoing that it is not only the integration algorithm but also how the results are sequenced that affect the results.

The second algorithm is treated is the popular second order Adams-Bashforth. Since this algorithm is second order, it implies that the derivative is approximated by a second order polynomial over some time interval. Hence, two past values of the derivative are required.

$$x_{n+1} = x_n + \frac{\Delta t}{12} (23\dot{x}_n - 16\dot{x}_{n-1} + 5\dot{x}_{n-2}) \quad (7)$$

It can be seen that this algorithm has two more terms than the Euler method which obviously leads to greater computational complexity.

The third category is the sinusoidal method developed by Rolston (1983). This algorithm was conceived specifically for aircraft simulation on the basis that aircraft dynamics are oscillatory and, therefore, a sinusoidal approximation of the derivative would be more accurate than some other function. Rolston developed two forms of the sinu-

soidal integrator. The sinusoidal predictor form is

$$x_{n+1} = x_n + P_o\dot{x}_n + P_1\dot{x}_{n-1} + P_2\dot{x}_{n-2} \quad (8)$$

where,

$$P_o = \frac{\omega_n\Delta t \sin(\omega_n\Delta t) - 2[1 - \cos(\omega_n\Delta t)][1 - 2\sin^2(\omega_n\Delta t)]}{2\omega_n \sin(\omega_n\Delta t)[1 - \cos(\omega_n\Delta t)]}$$

$$P_1 = \frac{-\omega_n\Delta t \cos(\omega_n\Delta t) - \sin(\omega_n\Delta t)[1 - 2\sin^2(\omega_n\Delta t)]}{\omega_n[1 - \cos(\omega_n\Delta t)]}$$

$$P_2 = \frac{\omega_n\Delta t \sin(\omega_n\Delta t) - 2\cos(\omega_n\Delta t)[\cos(\omega_n\Delta t - 1)]}{2\omega_n \sin(\omega_n\Delta t)[1 - \cos(\omega_n\Delta t)]}$$

The second form involves a sinusoidal extrapolator of the derivative which is integrated with a sinusoidal corrector. The extrapolator is given by

$$x_{n+1} = [1 + 2\cos(\omega_n\Delta t)](x_n - x_{n-1}) + x_{n-2} \quad (9)$$

The corrector form of the sinusoidal integrator is

$$x_n = x_{n-1} + C_o\dot{x}_n + C_1\dot{x}_{n-1} + C_2\dot{x}_{n-2} \quad (10)$$

where,

$$C_o = \frac{\omega_n\Delta t \sin(\omega_n\Delta t) - 2\cos(\omega_n\Delta t)[1 - \cos(\omega_n\Delta t)]}{2\omega_n\sin(\omega_n\Delta t)[1 - \cos(\omega_n\Delta t)]}$$

$$C_1 = \frac{-\omega_n\Delta t \cos(\omega_n\Delta t) + \sin(\omega_n\Delta t)}{\omega_n[1 - \cos(\omega_n\Delta t)]}$$

$$C_2 = \frac{\omega_n\Delta t \sin(\omega_n\Delta t) - 2[1 - \cos(\omega_n\Delta t)]}{2\omega_n \sin(\omega_n\Delta t)[1 - \cos(\omega_n\Delta t)]}$$

The final integration method presented in this paper is the Howe modified Euler technique developed by Howe (1989).

First the velocity at the half frame interval is found from

$$\dot{X}_{n+1/2} = \dot{X}_{n-1} + \Delta t\ddot{X}_n \quad (11)$$

then

$$\dot{x}'_{n+1} = \dot{x}_{n+1/2} + \Delta t(0.875\ddot{x}_n - 0.375\ddot{x}_{n-1}) \quad (12)$$

the second integration which yields displacement at the integer frame interval, is found from

$$X_{n+1} = X_n + \Delta t\dot{X}_{n+1/2} \quad (13)$$

then

$$x'_{n+3/2} = x_{n+1} + \Delta t(0.875\dot{x}_{n+1/2} - 0.375\dot{x}_{n-1/2}) \quad (14)$$

The primed values of equations (12) and (14) are used in the aerodynamic force and moment equations.

As was stated previously the aircraft simulation is a full force and moment, non-linear, six degree of freedom implementation. The aerodynamic force and moment are computed from a buildup of the six force and moment coefficients C_D, C_Y, C_L, C_l, C_m, C_n which are used with constant mass properties to compute the six acceleration components along and about the aircraft body axes. These accelerations are subsequently integrated to yield velocity components. First, the rotational components are determined; from which the quaternion rates are obtained which are then integrated to yield the quaternions. The quaternions are then used to obtain the direction cosines defining the relationship between the aircraft body axes and the earth fixed reference axis system. These direction cosines are used to obtain the aircraft orientation with respect to the earth axis system and to project the translational velocity components into the earth axis system whereupon they are integrated to give the aircraft position with respect to the earth. It should be noted that proper sequencing of these calculations is essential to ensuring minimum quantization effects.

Results

First, the results of the damped sinusoid dynamics will be discussed. Figure 1. presents the results of integrating equation (2) once to obtain velocity and then again to obtain displacement, each plotted for ten seconds. This is sufficient time for the system to be approaching steady state conditions. Two integration algorithms

are employed, the Adams-Bashforth second order (AB-2) updated at 5Hz (dotted curve), advanced Euler at 5Hz (dashed curve) and advanced Euler at 20Hz (solid curve). Notice that the 20 Hz advanced
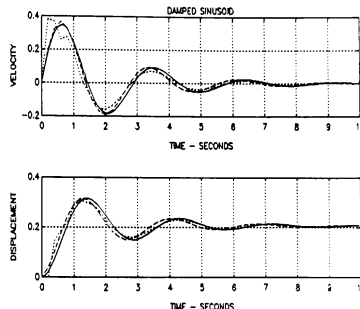

Figure 1. Comparison of AB-2 at 5 Hz(....) with advanced Euler at both 5Hz (----) and 20 Hz (____) for a second order system step response.

Euler seems to provide the smoothest response which is what is expected. However, what might not be expected is that the 5Hz advanced Euler provides smoother results than AB-2. It is believed that the fairly severe discontinuities in the first second are do to the effect of the two past values of the derivative being zero at t = 0. It can also be seen in the displacement plot that there is a one frame delay in the AB-2 solution with respect to the advanced Euler. This effect is more evident in figure 2. which is essentially the same as figure 1. except that it is plotted for two seconds to illuminate the initial response.
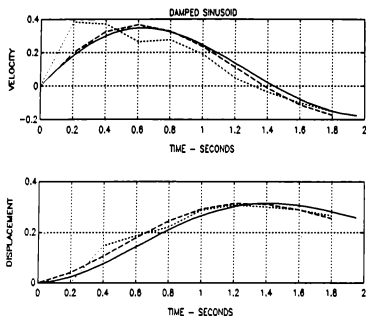
Figure 3. shows the results of all

Figure 2. Same as figure 1. except duration of plot is 2 seconds.

AB-2 displacement which is not clear in the plot. This delay is not present in either the advanced Euler, the Howe modified Euler or the sinusoidal methods.
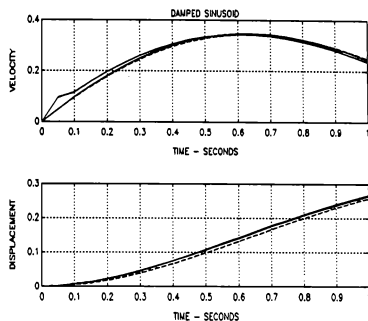

Figure 3. Comparison of all five integration algorithms at 20Hz and a continuous solution for the second order step response.

five integration algorithms executed at 20Hz and a "continuous" solution computed by MATLAB. It can be seen in the velocity plot that the AB-2, the sinusoidal predictor, the sinusoidal corrector and the Howe modified Euler all lie nearly on top of one another and have a knee at 0.05 seconds. This is due to the initialization of the past values of the derivative to zero at t = 0. The other curve is the advanced Euler and the continuous almost on top of each other. Since advanced Euler does not use past values of the derivative, there is no knee.

The displacement curve shows all five methods essentially superimposed on one another and the continuous (dashed line) alone. Another interesting fact is that there is a one frame delay in the

Figure 4. illustrates the effects of the various integration algorithms when they are employed to perform the integration required in the equations of motion simulation of a T-38 aircraft. The bottom set of curves is the simulated aircraft pitch rate and the top is the simulated aircraft angle of attack. In this case the advanced Euler, the Howe modified Euler and the sinusoidal predictor all yield essentially the same dynamics (solid lines). The AB-2 is the dashed line which deviates from the others slightly and the sinusoidal corrector (dot-dashed lines) which deviates greatly from the others. The behavior of the sinusoidal corrector is not consistent with the second order response cases previously. There may be an error in the implementation which was not discovered by press time. Once aircraft dynam
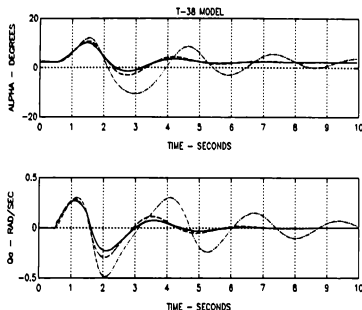
Figure 4. T-38 aircraft dynamics employing all five integration algorithms at 20Hz.

Figure 5. T-38 Aircraft dynamics employing all five integration algorithms at 10Hz.

ics are involved the picture gets a bit convoluted because the results depend greatly on the manner and sequence of calculations. In this case, the pitch rate (Q_A) is computed by integrating the pitch acceleration which is derived directly from the moment equation. However, the angle of attack is computed from translational velocity components. Therefore, the angle is not a direct result of two integrations.

All five of the algorithms are again presented in figure 5. which is similar to figure 4. with the exception that all the algorithms are executed at 10Hz. The two solid line curves are the advanced Euler and the Howe modified Euler. The dashed line is the AB-2 curve, the dotted line is the sinusoidal predictor and the dash-dot line is the sinusoidal corrector method. The velocity (Q_A) curves show more computational anomalies than the angle of attack curves particularly in the case of the sinusoi-
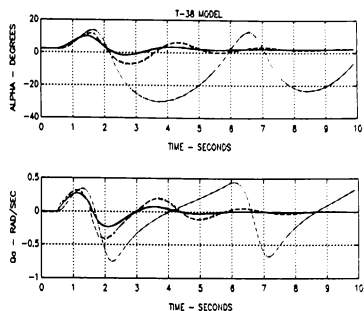
dal corrector. The AB-2 sinusoidal predictor display the characteristic knee which is a consequence of the past values of the derivative being zero when the aircraft is in trim. In the angle of attack curves the initial phase lead benefit of the AB-2 and Howe modified Euler is evident. However, this lead seems to dissipate after about two seconds in the angle of attack. This is due to the manner in which angle of attack is calculated. The persistent phase lead exchibited by these algorithms in the pitch rate plots would be observable in the aircraft pitch angle.

Considerable phase differences are illustrated in these curves among the various methods.

Figure 6. presents the first two seconds of the data of figure 4., namely, all five algorithms executed at 20Hz. In this format the lead associated with the two Euler methods and also the sinusoidal predictor (solid lines) is observable where it was not in figure 4.

The AB-2 method is the dashed line and the sinusoidal corrector is the dot-dash curve.

To demonstrate the effect of update rate on simulated aircraft response, angle of attack is computed employing advanced Euler executed at 20Hz, 50Hz and 100Hz and AB-2 at 20Hz and 50Hz. These results are presented in figure 7. where the first second of the response is plotted. The first half second is the period during which the simulated aircraft is reaching its trim point. At the 0.5 second point the elevator
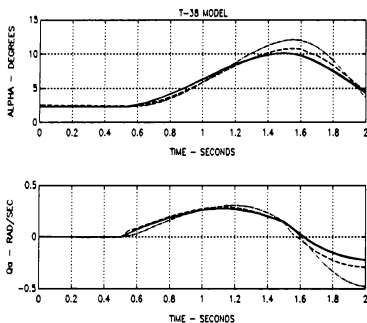

Figure 6. Comparison of all five methods executed at 20Hz with only the first two seconds plotted.

step is inserted and the simulated aircraft responds. To identify the various curves read from left to right and in order are advanced Euler at 20Hz, 50Hz and 100Hz, AB-2 at 50Hz and 20Hz. The 100Hz advanced Euler should be the closest to a continuous solution.

Conclusions

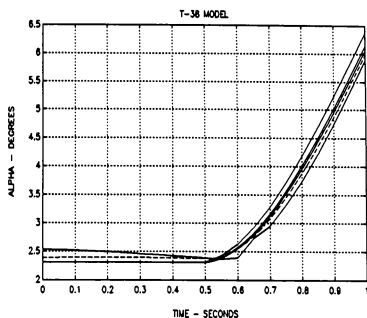At execution rates which are rel-

Figure 7. Comparison of advanced Euler at 20Hz, 50Hz, and 100Hz with AB-2 at 20Hz and 50Hz for the simulated T-38 aircraft.

atively high with respect to the aircraft modes virtually all of the algorithms performed adequately with the exception of the sinusoidal corrector. But the poor performance of that algorithm in the aircraft dynamics is probably due to an implementation problem since it was not manifested in the second order system response. Since the aircraft mode which is being stimulated is approximately 0.4Hz an update rate of 20Hz is relatively high. Modern flight simulators generally execute the simulated aircraft equations of motion at 60Hz not because it is necessary for faithful representation of the aircraft dynamics but rather to permit synchronous operation with computer image generation systems (CIG) which operate at video rates.

The two Euler methods performed well and in addition provide lead which can be used to advantage in compensating some of the transport delay inherent in CIG systems.

In view of the relative perfor-
mance exhibited by the various
methods the optimum choice seems
to be the advanced Euler method.
The performance of both Euler
methods is essentially equivalent
but the advanced Euler method is
computationally less complex.
Both Euler methods are substan-
tially less complex than any of
the other methods. It was also
illustrated that advanced Euler at
20Hz produced results very close
to the continuous solution for the
second order system response with
essentially the same dynamics as
the T-38 longitudinal mode inves-
tigated.

It should also be noted that all
the higher order algorithms intro-
duce some discontinuities into the
early part of the solution. This
could be especially problematic
since motion system cuing algo-
rithms frequently use both air-
craft pitch rate and pitch angle.
These discontinuities could pro-
duce acceleration spikes in the
motion system and perhaps stepping
in the visual system. It was also
found that sequencing of the com-
putations is vital in providing
appropriate results since the
digital computer is serially cal-
culating what is actually a paral-
lel process represented by six
differential equations.

References

Howe, R.M. "An improved numerical
integration method for flight
simulation." Paper no. 89-3306
AIAA Flight Simulation Technolo-
gies Conference. AIAA, Washing-
ton, D.C. August 1989, pp. 310-
316.

Rolston, D.R. "Sinusoidal
integration for simulation of
second-order systems." Paper no.
83-1086, AIAA Flight Simulation
Technologies Conference, AIAA
Washington, D.C. 1983, pp. 52-63.

THE EFFECTS OF OBJECT ORIENTED DESIGN AND Ada ON THE MODELING ENVIRONMENT

Jerry O'Grady
Barbara A. Bowen

Simulation /Trainer Products, Grumman Corp.
Electronics Systems Division
10 Orville Dr.
Bohemia, N.Y.11716

ABSTRACT

Recent focus on Ada and Object-Oriented Design has induced a wide range of changes to the Software Engineering discipline. However, are these changes limited to software or are other disciplines affected? This paper will focus on the discoveries made by a Grumman Math Modeling group while developing trainer math models. These models were implemented in Ada and an Object-Oriented Design (OOD) environment. This will include:

Presentation of Math Models - how Ada and OOD can be used to clarify the presentation of a math model report through the use of improved readability and meaningful naming conventions.

Enhancement of the Math Model Content - improvements obtained through consideration of all trainer states, consideration of default conditions, accounting for the iteration rate as a variable and symbol dictionary improvements.

Math Model Development - through the use of software and mathematical tools, design validation can occur at an earlier stage. Included is a brief discussion on how a math model is developed and its effects on reusability from a modeling viewpoint.

Perceptions on Trainer CDRL Documentation - the perceptions presented will include the scheduling problems encountered with the required CDRL documentation.

Conclusions and lessons learned from Grumman's successful Ada program are included.

INTRODUCTION

In January 1988, Grumman Simulator/Trainer Products was awarded the an Ada Feasibility Study Contract. This contract encompassed re-hosting the existing computer system, hardware modifications, software and system enhancements. The software re-host and system enhancements were accomplished using a modified Object Oriented Design approach in the Ada software environment.

The performance baseline for the contract was defined as the Assembly Source and Object Code. The authors were tasked with the problem of converting the assembly code with all its subtleties into approximately thirty-four mathematical models. The models were to retain the existing trainer performance while allowing the software discipline the latitude of developing clear, concise, and efficient Ada code. The coded models were tested and then integrated using Grumman's test philosophy.

Grumman found that conforming to the Object Oriented Design and Ada forced a change in our traditional views of System Engineering/Math Modelling. The focus of this paper will be to share and discuss the presentation of Grumman's math models, the math model development cycle in the Ada environment, and schedule requirements. Conclusions with lessons learned from the project are provided.

The Positive Effects on Modeling

Presentation of Math Models

The presentation of Grumman's math model changed dramatically with Object Oriented Design. The effects of Ada and OOD on naming conventions, readability and presentation of the math models in the form of logic tables and equations resulted in a report which is easier for **anyone** to follow. The models are in a more generic format, thus allowing them to be reused on different projects with a minimal amount of rework.

The change in the naming convention had the most noticeable impact. Normally, mnemonic spelling conventions are established to represent a variable while providing the greatest amount of information possible. For example, a mnemonic may be defined by six characters. The first character describes the type of variable, the second and third denote the model in which the variable originates and the remaining characters further identify the variable. For instance, BAEDCV might represent DC voltmeter output where B tells the reader that the value is an analog output, AE implies an electrical system value and DCV is the modeler's abbreviation for DC voltmeter.

Ada allows the use of full names to describe a variable. The math modeler can now create a name which can be directly transported to Ada code. This name can easily be spotted and understood when reviewing or troubleshooting the Ada listing. The mnemonic example presented above would read:

ELECTRICAL_POWER_SYSTEM.OUTPUT_OF.DC_VOLTMETER

Where:

ELECTRICAL_POWER_SYSTEM = Package Name
OUTPUT_OF = Record Name
DC_VOLTMETER = Record Item Name

As you can see, the Ada name is much easier to understand without the aid (and thought interruptions) of a symbol dictionary. Even if renames are provided, identifying the variable is not a problem.

EPS = ELECTRICAL_POWER_SYSTEM.OUTPUT_OF

changes the above example to:

EPS.DC_VOLTMETER.

A tremendous amount of time can be saved during the initial creation, testing, modification and review (in-house and customer reviews), because the reader can adapt easily to the naming convention.

There are several methods for presenting a math model (i.e. equations, flow charts, PDL, truth tables). The modelers chose to use truth tables and equations. Regardless of the method chosen, we must ensure that language-peculiar logic not appear in the Math Model Report. In the past many math models turned into nothing more than flow chart representations of the listings.

By using truth tables, several benefits are gained. It creates a clearer picture of what is being developed by separating each output into a separate table. Truth tables force the modeler to create concise information. It is easy to lapse into complex multi-path logic using flow charts. Confusion as to where one calculation ends and the next begins occurs when logic is overlapped in flowchart. Flow charts can include direction ("go to's") which may cause oversights in calculations. Also, flow charts force direction upon the software engineer.

Truth tables and equations provide a concise, straight forward model, free from constraints on the program designer. For example, a flow chart may force an if/then/else statement when a case could be used. Logic tables would present only the conditions and the appropriate results allowing the software engineer creative freedom in choosing a method of implementation.

The problems associated with readability become a minute issue because of Ada. Creating variable names in the math model report that can be directly transferred to code provides three benefits.

1) The math model report becomes easier to read.

2) Working with the math model report and the listings to decipher the logic is simpler. This is because the transition from system to software is exact, the names are descriptive, self explanatory and the math modeler or software engineer will not rely so heavily on a symbol dictionary and comments. This is a great advantage since there are many occasions when the documents are not all completely updated.

3) If the method of presenting the math model report is chosen to be flow charts, it will then be easier to follow the math model and the listing than it has been in the past.

Enhancement of Math Model Content

Math Model content can be enhanced by considering the trainer's operating state, required default conditions, defining time as a variable, and by focusing on the model content provided in an Ada and OOD environment. This additional information provided by the modelers more accurately defines a complete detail design.

Grumman has chosen to break its trainer states into separate procedures. This is more object oriented. The designer is forced to pay more attention to the trainer's other states instead of defining them as an after thought. This often happens when freeze or reset logic is elsewhere or not emphasized. In using this methodology, it was necessary to lay out the math model differently than it has been done in the past.

After a brief System Description/Introduction, the majority of the model is detailed in the section titled RUN STATE. FREEZE STATE and RESET STATE followed with only the deviations from RUN STATE described in detail. For example, if the radio models are to operate during freeze, the FREEZE STATE processing should state run the same procedures as the run state, along with any initialization peculiarities.

This method of defining model states avoids confusion to all concerned, particularly to the modeler who would have to lace the model with freeze and reset considerations. Elimination of interlaced logic provides for a cleaner presentation.

Our technique gives each math model a default IC-namely "On the Ground, Engines Off, APU connected; fuel tanks full; all malfunctions disabled." This allows two things to happen:

1) All Ada variables will be capable of being set to a default value.

2) The trainer can be tested without the IOS initial condition set up in place.

We recommend that "time" be the determining factor in how long a condition lasts, not frame count. If the system iteration rate was to change, a transitory condition based on a frame count could be inadvertently modified. One based on DELTA TIME would not.

Defining iteration rate as a variable can reduce modification time if the iteration rate were to change. Previously, iteration rates were combined with other constants. The new value formed was code and time efficient but often not adequately defined. Also, the iteration rate, when defined as a constant, could easily be overlooked if the update rate were modified. By breaking time out as a variable, two advantages are then obtained. Modification of the model is simplified because time is well defined and not combined or hidden within other constants. The second advantage is that time can be easily updated by changing only the area that defines the iteration rates (reference Figure 1).

A change to the fuel quantity drop rate need only by done once in the local variable list and all areas in the listing that use the variable will automatically be modified from:

Fuel quantity drop rate : = lbs/sec : = 100;

to:

Fuel quantity drop rate : = lbs/sec : = 80;

Also by defining time as a variable a change in iteration rate can be made quicker by simply redefining delta-time from:

delta-time   : = seconds   : = 0.0625;

to:

delta-time   : = seconds   : = 0.06;

After the math modeler defines names, units, range and mathematical symbol for the initial cut of the symbol dictionary the software tools in the Ada environment can be used to generate the symbol dictionary. Grumman decided that the Software Documents and Math Model Report would use a common symbol dictionary. The common dictionary saves the modeler the time normally spent generating and maintaining his own dictionary. The most important features are the commonality and maintainability that is provided between the Math Models and the Software.

Where:

6.25 = DROP IN FUEL QUANTITY PER PASS

(PREVIOUS METHOD)

FUEL QUANTITY PUSH-TO-TEST ENGAGED	FUEL QUANTITY
TRUE	FUEL QUANTITY -Δ t • FUEL QUANTITY DROP RATE |,

Where:

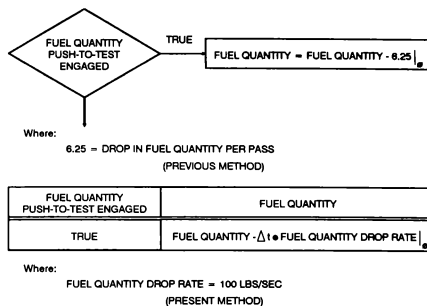FUEL QUANTITY DROP RATE = 100 LBS/SEC

(PRESENT METHOD)

Figure 1 Delta Time Comparison

The math model report should provide the following information to the Software Engineer, with the necessary details, in order to design maintainable Ada code:

UNITS: Ada type checking makes it necessary for the modeler to provide the units for each value defined. The modeler is forced to look more closely at the values being provided to the Software Engineer. As a result, troubleshooting time is eliminated later on.

CONSTANT NAMES: The modeler should provide in the Math Model Report a short name which describes a constant. This name can be implemented in the code instead of the actual literal. As previously shown, changes to the constant need only be done once in the local variable list. All areas in the listing that use implement the variable will automatically be modified.

RANGES: Although range is not required for the code to run, it is helpful to define the applicable range so during testing the out-of-tolerance conditions can be easily defined and tested.

DEFAULT VALUES: These initial run defaults should be defined so the message specifications can include them. This removes the chance of arithmetic errors occurring during initialization of the device.

It is suggested that a working symbol dictionary (See Figure 2) be created by the math modeler to define the information described.

Math Model Development

The Grumman math models benefited from Ada and OOD development environment. These benefits appeared in the form of earlier validation, improved and new model development tools, and the possibility of reusing a model With these developments which are

SYMBOL	NAME ASSIGNMENT	I/O	G/L	UNITS	RANGE	DEFAULT
N	LATITUDE	O	G	DEGREES	[-60,60]	31.0
H	FIELD HEIGHT	O	G	FEET	[-2000,32000]	298.0

Figure 2 Symbol Dictionary

described below, the modeler's task becomes easier, their work more precise and testing time shorter.

A general rule of thumb for time spent on a job is 40-20-40 where 40% of the time is spent on design, 20% on coding and 40% during HSI. The drawback is that most of the time is spent during the expensive part of the job (Remember, the later problems are caught, the more expensive they become to fix).

The Ada environment requires a tremendous amount of up front work. At first this may seem tedious, but the time spent doing this work helps eliminate integration problems down the line. For example, defining all units early in development may be time consuming, but Ada will check these units through type checking to ensure proper use. Any interface mismatches will be flagged during PDL development.

Ada shifted the job time split to 50%-20%-30% there by moving 10% of work time out of the more costly HSI time frame.

Use of certain math models tools also allow for earlier validation possibilities. One of these tools, Software System Design's "Testgen", was particularly helpful. It is described in the following paragraphs.

During development, the mathematical model is developed in Ada PDL, unit tested comparing the PDL generated Testgen results and the math model, coded, retested using Testgen, and integrated into the design. In parallel with the generation of the source code, the unit test procedures are produced. The production software is then submitted to static checks (compilation and code review) and dynamic checks (alpha/beta tests).

During the feasibility project, Grumman used Software Systems Design's "Testgen" tool. The tool, when invoked upon the program design language showed the math modeler early in the development cycle what flaws existed in the preliminary design. The modeler was then able to start generating the unit test procedures in an organized and planned way using the test cases defined by the tools "Design Review Expert Assistant". This task was achieved concurrently with the coding of the model. The "Testgen" tool was then used on the code to verify that the test cases generated for the PDL are still valid for the executable Ada code. The tool also verifies that all instructions can be reached. The

"Testgen" tool proved to be invaluable up to this point. The tools other features: test driver, test cover analyzer and strategy generator were immature and could not be used to accomplish dynamic testing. Grumman provided a workaround to this problem by developing its own tools to accomplish dynamic testing.

An alpha test is an off-line test (non-real time) of each Ada package(Object) independent of other packages. Each package is alpha (or unit) tested to verify the following:

• All inputs to the package are correctly interpreted.

• Arithmetic and logical functions and or sub-modules assigned to the package are correctly processed.

• Outputs are correct and consistent with the input data.

• All paths in the package are traversed.

Alpha testing was accomplished using a Grumman developed configured test driver installed on the Micro Vax II development computer. The test driver determined the paths and/or branches that were exercised and recorded the results of the test.

After each Object successfully passed alpha testing, it was then subjected to Beta Testing, where possible. A beta test validates interfaces, in real time, with two or more alpha-tested packages and verifies compatibility. Each group of packages is beta tested to verify the following:

• Interfaces between packages are correct.

• Scope of definition of variables is unambiguous.

• The packages operate as designed and according to design requirements.

Beta testing was accomplished using a programmable interactive test set developed by Grumman. The test set provided the capability of integrating groups of models together in real time. Testing was accomplished on the target computer in all trainer states, while not requiring the trainee station or IOS hardware to be present. This allowed the math modelers and the software engineers the ability to compress the integration schedule by testing at an earlier stage in the project. As each math model is successfully beta tested, it is then incorporated in the system control structure.

A mathematical model, developed with good engineering practice can be implemented in any

language, used with any executive or adapted to any design philosophy. Good engineering practice can be defined as:

- All definitions provided

- All ranges and units defined

- Assumptions stated clearly

- Input/Process/Output diagram provided

- Valid iteration rates

- Freezer & reset considerations

- All symbology defined

The question brought forth is, is the software that is developed from the math model really reusable? The answer is not necessarily yes, however, it is not definitively no. There are too many variables that must enter into the discussion. Let's look at two major points. The executive that is developed by each simulator manufacturer can be different and the idiosyncracies that each executive exhibits will require an adaptation/modification to the software object to be reusable. The "types" selected by different simulator manufacturers' could cause major software modifications and testing to reuse a software object. There are many possible ways of selecting types. Two possibilities are shown below, Table 1 defined a class of units and Table 2 defined actual units. With the above problems in mind, the software and its supporting documentation is not really reusable between the manufacturers. The example has similar conversion problems that one would encounter if you tried to reuse models developed in Fortran.

The Adverse Effects on Modeling

Presentation of Math Models

Although generally, OOD enhances a math model report, there is a minor drawback.

type Pressure	is digits 6 range 0.0 .. 10000.0; - pounds per square inch
type Temperature	is range 300 .. 3000; - degrees Rankine
type Air_Flow	is digits 4 range 0.0 .. 500.0; - pounds per second
type Thrust	is digits 6 range 0.0 .. 20250.0; - pounds
type RPM	is range 0 .. 20000; - revolutions per minute
type Torque	is range 0 .. 10000; - pound feet

Table 1 Type is a Class of Units

```
ACCELERATION:
   type FEET_PER_SECOND_SQUARED is new STANDARD_FLOAT;
FORCE:
   type POUNDS is new STANDARD_FLOAT;
   type G_s is new STANDARD_FLOAT;
MASS:
   type POUNDS_MASS is new STANDARD_FLOAT;
   type SLUGS is new STANDARD_FLOAT;
ELECTRICAL
   type AMPS is new STANDARD_FLOAT digits 4;
   type VOLTS is new STANDARD_FLOAT;
ANGULAR
   type RADIAN is new STANDARD_FLOAT;
   type RADIANS_PER_SECOND is new STANDARD_FLOAT;
   type RADIANS_PER_SECOND_SQUARED is new
   STANDARD_FLOAT;
```

Table 2 Type is Actual Units

We have seen how Ada naming conventions can positively effect a math model report. The only real drawback to this is the length of the names. In the example given earlier, this is not evident, but let's take a different example. Body Axis Longitudinal Velocity is quite an awkward name, especially when EQUATIONS OF MOTION.OUTPUT OF is tagged onto the front of it. A modeler using a standard symbol would call this U and greatly shorten the name. But now, does U mean anything to the software engineer? How detailed should the names be?

Lessons Learned

1) The use of short descriptive names improves readability, maintainability and provides the reader with the capability to reference the math model and software documents without trying to juggle the symbol dictionary as well.

2) Logic Table and Equations present all possible solutions. They present one result per table and provide a method of presentation that is free of language constraints.

3) Delta Time defined as a variable provides clarity and improves maintainability.

4) Presenting the trainer states individually emphasizes any special needs that previously, were not required by the math model "DID".

Perceptions of CDRL Document Requirements

During the UH-1 Project Grumman was queried on the need for the system documents and the order in which the data should be delivered. The resulting perceptions on these areas are presented for you to digest.

There has been discussion by the customer of reducing the number of deliverable documents to reduce the costs associated with document generation, change and maintainability.

Some of the documents probably should be deleted from the CDRL list. However, each document on the CDRL list had a specific purpose intended for it. This brings us to the issue of the "System Documents". There is a tendency today to wonder why the contractor needs to deliver a Math Model Report or a Criteria Report. The feeling is that with Ada and OOD as the design methodology of choice, the reusable software and the documents that support it will be sufficient to maintain the simulator through its life cycle. The fact is that the SRS and SDD are not sufficient and they are written to support the detailed requirements and design of particular computer software configuration item. These documents although complete and specific to how the software is supposed to perform lack the basic foundation for developing and understanding a mathematical model.

If the government is to rely on only the Software Requirements Specification and delete the Criteria Report the authors pose the following questions:

1) Where is the prime system criteria to be defined?

2) Where are the system details for the malfunctions to be defined?

3) Where are the simulated system tolerances to be defined?

4) Where are the flying qualities and performance characteristics going to be presented?

5) What document will provide the source for TTPRR development since the SRS does not present the information in question 1 through 4. The SRS is a software requirement specification where as the Criteria Report is suppose to represent the entire integrated system, hardware and software.

A similar set of questions can be asked if the government relied only on the Software Detail Design Document and deleted the Math Model Report.

1) Where would the mathematical assumptions and simplifications be discussed?

2) Where would the error analysis be presented?

3) Where will the SDD obtain the model to be used?

The authors do note that documentation costs are increasing and that measures should be taken to curtail the escalation. The authors also acknowledge that there is a great deal of duplication of effort between the modelling environment and software development. This duplication of effort should be removed and the pertinent design considerations retained. Deletion of the system documents without consideration to the above questions would prove to be an expensive lesson later in the life cycle of a flight trainer. Before a decision for future contracts is made the Data Item Descriptions for the system documents and software documents should be reviewed for completeness. They must then be refined in such a way that extraneous information is no longer required. However, pertinent and important design information is retained.

Hardware, Software and System engineering are the critical disciplines in the development of a flight trainer. The integration of the three disciplines is necessary if the requirements of the user and the successful completion of the contract are to be achieved. The availability of engineering data is shown in figure 3. This availability reflects the parallel engineering efforts encountered by the disciplines in the required sequence. The Data Item Description's should be tailored to minimize any duplication of effort. The schedule shown reflects when the data is available and not when the actual document is due for submittal. The complexity of the trainer and constraints of the contract should determine when documents are submitted and the required number of submittals per document.

Conclusions

1) There is a need for mature software tools that can support the complex real-time simulation environment.

2) Mathematical models designed to good engineering practice are reusable. The structural model and design philosophy that each simulator manufacturer imparts on the developed code does not guarantee compatibility between two manufacturers. Reusability may not be attainable at the software level unless standardized interfaces are required.

3) It is not in the customers best interest to delete the system documents from the CDRL list. The DID's for the system documents should be reviewed for completeness and refined. This way costs are cut and the customer retains all pertinent data.
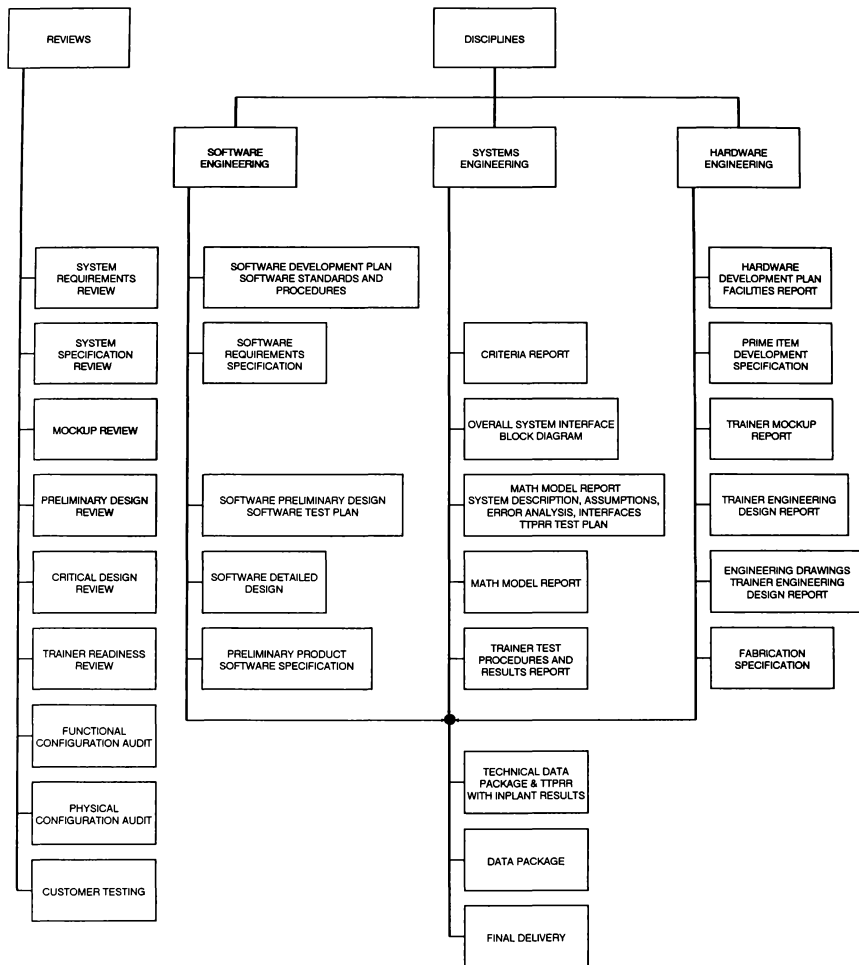
ACKNOWLEDGEMENTS

Figure 3 Engineering Data Availability

BIBLIOGRAPHY

1) UH-1 FS Ada Feasibility Study Project, Lessons Learned Briefing 9 Feb., 1989, Contract N61339-88-0010 CDRL H003 Conference Minutes.

2) DOD-STD-2167. Defense System Software Development.

3) DI-MCCR-80025. Software Requirements Specification.

4) DI-MCCR-80031. Software Detail Design Document.

5) UDI-S-25589. Trainer Criteria Report.

6) UDI-S-25592A. Trainer Math Model Report.

7) UDI-T-25594C. Trainer Test Procedures and Results Report.

ABOUT THE AUTHORS

Jerry O'Grady is a Senior Systems Engineer with Grumman Electronics Systems Division in the Simulation/Trainer Products group. He is currently the Lead Systems Engineer for the UH-1 FS Ada Feasibility Study. Before this assignment, Mr. O'Grady has been responsible for development and integration of various flight and system models for the following Weapon System Trainers: SH-60F, A-6E and F-14A. Mr. O'Grady has nine years of Simulation experience. He holds a Bachelor of Science Degree in Aeronautical Engineering from Polytechnic Institute of N.Y., Brooklyn, N.Y. and has recently completed his Masters Degree at Adelphi University, Garden City, N.Y..

Barbara Bowen is a Systems Engineer with Grumman Electronics Systems Division in the Simulation/Trainer Products group. She is currently developing, testing and integrating math models for the UH-1 FS Ada Feasibility Study. Before this assignment, Ms. Bowen developed math models and performed updates to the F-14 Operational Flight Trainer and F-14 Weapon System Trainer. Ms. Bowen has been a Math Modeler for six years and holds a B.S. and M.S. in Pure Mathematics from SUNY at Stony Brook and Adelphi University, respectively.

A NUMERICAL TECHNIQUE
FOR SOLVING THE COUPLED-EQUATIONS OF
MOTION OF AN AIRPLANE

Muhammad Abdullah Ghazi*
Aeronautical Eng. Dept., King Abdulaziz University
P.O. Box 9027, Jeddah 21413, Saudi Arabia

Abstract

The fact is that the airplane is indeed a complicated dynamic system. Many times it moves in several modes at the same time, nevertheless it is required to have a safe flight. Beginning with Lanchaster about the year 1900, many investigators have shown interest in the dynamic stability and response of airplanes. Of utmost importance in this regard is the theorem of small disturbances, which seems to hold promise for a practical solution.

Undoubtedly the response of an airplane to external disturbances is an intricate phenomenon, especially during maneuvering flight, when the coupling of longitudinal as well as lateral disturbances are coupled. The subject deserves the conduction of comprehensive research. In the present work we present a simple numerical technique for the solution of the coupled equations of motion. To this end the general non-dimensional small disturbance equations are obtained as a system of first order linearized differential equations. The equations are written in vectorial form, and denoted by the state equation. The output parameters of this equation are continuous for finite inputs. It is therefore satisfactorily integrated by trapezoidal integration, which is stable for all values of the increment size. For discontinuous excitation, as in control inputs, the use of rectangular integration provides the best approximation.

The Z-transform notation is utilized to obtain the recursion relation, which is the solution for the state equation, i.e., the equations of motion. The output of a computer program that is developed for the solution of this recursive relation provides an illustration for the airplane responses following initial disturbances during maneuvering or straight flight. Complete agreement is established between the stability criteria extracted from the response curves of the computer output, and results from earlier researches for both the longitudinal and the lateral dynamic stability during cruising flight. Samples from the computer outputs are presented.

Introduction

The equations of motion of a symmetrical airplane, for the case when motion is limited to infinitesimal disturbances from steady straight symmetric flight, have been extensively studied, beginning about the year 1900, with Lanchester.[1,2,3] Lanchester surveyed the ground with

unconventional mathematical methods, and was able to arrive at many of the broad conclusions now used in practical aerodynamics.

Bryan and Williams[4] made use of more conventional mathematical methods later to introduce the linearized equations of motion. The latter have been the foundations of studies of dynamic stability and response to control ever since. Bryan[5] then presented the theories of both the longitudinal and lateral motions. He also showed that the solutions could be made to depend on a number of experimentally certain constants or derivatives, known as stability derivatives.

Bairstow and others at the National Physical Laboratory (NPL) developed the experimental technique necessary to determine the stability derivatives. They demonstrated how essential features of practical interest could be extracted from the rather elaborate solution that arises even under these restricted circumstances.[6] Hunsaker[7], who had visited the NPL in 1914, introduced in the United States, Bairstow's wind tunnel techniques, and the method of Bryan and Bairstow for the computation of dynamic stability.

Glavert calculated the motion of an airplane with the elevator free, and introduced the non-dimensional form of the equations of motion.[8,9] Detailed applications of the equations of motion to practical problems were given by Roskam,[10] Robert Nelson[11] and Babister.[12] By 1935, when the survey by B. Melvill Jones appeared in Durand's Aerodynamic Theory, reprinted in 1976,[3] the classical approach of Bryan and Bairstow was well established but was very little used. Results of full-scale experiments had led to the conviction that the theory of infinitesimal motions was practical for the prediction of the stability of motion, the time-history of motion following a disturbance, and the response to the application of control.

Undoubtedly the response of an airplane to external disturbances is a complicated phenomenon, especially during maneuvering flight, when the longitudinal and lateral disturbances are coupled together. The subject deserves the conduction of comprehensive research. In the present work we present a simple numerical technique for the solution of the coupled equations of motion.

Nomenclature

A = moment of inertia about rolling axis, ox
B = moment of inertia about pitching axis, oy
b = wing span
C = moment of inertia about yawing axis, oz
C_L = lift coefficient

---

* Associate Professor, Aeronautical Engineering
  Member AIAA

c̄ =mean aerodynamic chord of wing
E =product of inertia about ox and oy
g =gravitational acceleration
i =non-dimensional moment of inertia, or non-dimensional product of inertia
L =rolling moment about ox
M =pitching moment about oy
m =airplane mass
N =yawing moment about oz
P =steady rolling velocity
p =rolling velocity disturbance
Q =steady pitching velocity
q =pitching velocity disturbance
R =steady yawing velocity
r =yawing velocity disturbance
S =wing area
u =forward velocity disturbance, or, control deflections
V =flight speed
v =side velocity disturbance
w =downward velocity disturbance
ζ =rudder deflection
η =elevator deflection
ξ =aileron deflection
θ =angle of pitch
ϕ =angle of bank
ψ =angle of yaw
μ_1 =longitudinal relative density parameter, $m/(0.5\rho S\bar{c})$
μ_2 =longitudinal relative density parameter, $m/(0.5\rho Sb)$
ρ =air density
τ =time unit, $m/(0.5\rho VS)$

Subscripts
n =non-dimensional derivative
o =initial
p =derivative with respect to p
q =derivative with respect to q
r =derivative with respect to r
u =derivative with respect to u
v =derivative with respect to v
w =derivative with respect to w
ẇ =derivative with respect to ẇ
x =about longitudinal (rolling) axis, i.e., x-axis
xz =about x-axis and z-axis
y =about lateral (pitching) axis, i.e., y-axis
z =about yawing axis, i.e., z-axis
ζ =derivative with respect to ζ
η =derivative with respect to η
ξ =derivative with respect to ξ

Superscripts
(^) =non-dimensional parameter
(˙) =first derivative

Equations of Motion and the Algorithm

The small disturbance equations of motion relative to the stability axis are written in the following form.

Linear momentum equations

$$m\dot{u}=(X_u)u+(mR_o)v+(X_w-mQ_o)w+(X_q)q+(-mg\cos\theta_o)\theta+(X_\bullet)\eta \quad [1]$$

$$m\dot{v}=(-mR_o)u+(Y_v)v+(mP_o)w+(Y_p)p+(Y_r-mU_o)r$$
$$+(-mg\sin\theta_o\cdot\sin\phi_o)\theta+(mg\cos\theta_o\cdot\cos\phi_o)\phi$$
$$+(Y_\xi)\xi+(Y_\zeta)\zeta \quad [2]$$

$$m\dot{w}=(Z_u+mQ_o)u+(-mP_o)v+(Z_w)w+(Z_q)q$$
$$+(-mg\sin\theta_o\cdot\cos\phi_o)\theta+(-mg\cos\theta_o\cdot\sin\phi_o)\phi+(Z_\eta)\eta \quad [3]$$

Angular momentum equations

$$(CA-E^2)\dot{p}=(CL_v+EN_v)v+\{(CL_p+EN_p)-EQ_o(B-A-C)\}p$$
$$+\{-C(CR_o-EP_o-BR_o)-E(BP_o-AP_o+ER_o)\}q$$
$$+\{(CL_r+EN_r)-Q_o(E^2+C^2-CB)\}r+\{C(L_\xi\cdot\xi+L_\zeta\cdot\zeta)$$
$$+E(N_\xi\cdot\xi+L_\zeta\zeta)\} \quad [4]$$

$$B\dot{q}=\{M._u+M_{\dot{w}}^\bullet((Z_u/m)+Q_o)\}u+(-M_{\dot{w}}^\bullet P_o)v+\{M_w$$
$$+(M_{\dot{w}}^\bullet Z_w/m)\}w+\{-(AR_o+2EP_o-CR_o)\}p+\{M_q$$
$$+M_{\dot{w}}^\bullet((Z_q/m)+U_o)\}q+\{-(AP_o-2ER_o-CP_o)\}r$$
$$+\{-M_{\dot{w}}^\bullet g\sin\theta_o\cdot\cos\phi_o\}\theta+\{-M_{\dot{w}}^\bullet g\cos\theta_o\cdot\sin\phi_o\}\phi$$
$$+\{M_\eta+(M_{\dot{w}}^\bullet Z_\eta/m)\}\eta \quad [5]$$

$$(CA-E^2)\dot{r}=\{EL_v+AN_v\}v+\{(EL_p+AN_p)-Q_o[(B-A)A$$
$$-E^2]\}p+\{-A(BP_o-AP_o+ER_o)-E(CR_o-EP_o-BR_o)\}q$$
$$+\{(EL_r+AN_r)-Q_oE[A+(C-B)]\}r+[E(L_\xi\,\xi+L_\zeta\,\zeta)$$
$$+A(N_\xi\,\xi+N_\zeta\,\zeta)] \quad [6]$$

Orientation Equations

$$\dot{\theta}=(\cos\phi_o)q+(-\sin\phi_o)r+(-Q_o\sin\phi_o-R_o\cos\phi_o)\phi \quad [7]$$

$$\dot{\phi}=p+(\tan\theta_o\cdot\sin\phi_o)q+(\tan\theta_o\cdot\cos\phi_o)r$$
$$+\{(Q_o\sin\phi_o+R_o\cos\phi_o)\sec^2\theta_o\}\theta$$
$$+(Q_o\tan\theta_o\cdot\cos\phi_o-R_o\tan\theta_o\cdot\sin\phi_o)\phi \quad [8]$$

$$\dot{\psi}=(\sec\theta_o\cdot\sin\phi_o)q+(\sec\theta_o\cdot\cos\phi_o)r+\{\tan\theta_o\cdot\sec\theta_o(Q_o\sin\phi_o$$
$$+R_o\cos\phi_o)\}\theta+\{\sec\theta_o\cdot(Q_o\cos\phi_o-R_o\sin\phi_o)\}\phi \quad [9]$$

Reference steady state conditions for these equations are stated as follows:

1. Linear velocities and accelerations
   * the linear velocity in the x-direction is U_o

* the linear velocity in the y-direction is zero
* the linear velocity in the z-direction is zero
* the linear accelerations in the x, y and z-directions vanish.
2. Angular velocities and accelerations
    * angular velocity about the x-axis is P_o
    * angular velocity about the y-axis is Q_o
    * angular velocity about the z-axis is R_o
    * angular accelerations about the x, y, and z-axes vanish.
3. Orientation angles
    * the reference bank, elevation, and azimuth angles are Φ_o, θ_o, and ψ_o, respectively.

The above equations of motion are expressed in non-dimensional form in accordance with the notations of Babister.[12] The resulting relations comprise a system of first order linearized non-dimensional equations.

Non-dimensional linear momentum equations

$$\hat{\dot{u}} = X_{u,n}\hat{u} + \hat{R}_o \ \hat{v} + (X_{w,n}-\hat{Q}_o)\hat{w} + (X_{q,n}/\mu_1)\hat{q}$$
$$+ (-\hat{g}_1)\theta + (X_{n,n}\eta) \tag{10}$$

$$\hat{\dot{v}} = (-\hat{R}_o)\hat{u} + (Y_{v,n})\hat{v} + (\hat{P}_o)\hat{w} + (Y_{p,n}/\mu_2)\hat{p} + ((Y_{r,n}/\mu_2) -1)\hat{r}$$
$$+ (-\hat{g}_2 \sin\Phi_o)\theta + (\hat{g}_1 \cos\Phi_o)\Phi + Y_{\xi,n}\xi + Y_\zeta\zeta \tag{11}$$

$$\hat{\dot{w}} = (Z_{u,n}+\hat{Q}_o)\hat{u} + (-\hat{P}_o)\hat{v} + (Z_{w,n})\hat{w} + (Z_{q,n}/\mu_1 + 1)\hat{q}$$
$$+ (-\hat{g}_2 \cos\Phi_o)\theta + (-\hat{g}_1 \sin\Phi_o)\Phi + Z_{n,n}\eta \tag{12}$$

Non-dimensional angular momentum equations

$$(i_x i_z - i^2_{xz}) \ \hat{\dot{p}} = [\mu_2(i_z L_{v,n} + i_{xz}N_{v,n})]\hat{v} + \{i_z L_{p,n} + i_{xz}N_{p,n}$$
$$+ \hat{Q}_o i_{xz}(i_x+i_z-i_y(\mu_2/\mu_1)^2)]\hat{p} + [i_{xz}\hat{P}_o(i_x-i_y(\mu_2/\mu_1)^2$$
$$+ i_z)+\hat{R}_o(i_z i_y(\mu_2/\mu_1)^2)-i^2_x-i^2_{xz})]\hat{q} + [i_z L_{r,n}+i_{xz}N_{r,n}$$
$$+ \hat{Q}_o(i_z i_y(\mu_2/\mu_1)^2)-i^2_{xz}-i^2_z)]\hat{r} + [\mu_2 i_z(L_{\xi,n}\xi+L_{\zeta,n}\zeta)$$
$$+ \mu_2 i_{xz}(N_{\xi,n}\xi+N_{\zeta,n}\zeta)] \tag{13}$$

$$(i_y/\mu_1) \ \hat{\dot{q}} = [M_{u,n}+(M^{\bullet}_{w,n}/\mu_1)(Z_{u,n}+\hat{Q}_o)]\hat{u}+[-(M^{\bullet}_{w,n}/\mu_1)\hat{P}_o]\hat{v}$$
$$+[M_{w,n}+(M^{\bullet}_{w,n}/\mu_1)Z_{w,n}]\hat{w} + \{[(\mu_1/\mu^2_2)(i_z-i_x)\hat{R}_o$$
$$- 2(\mu_1/\mu^2_2)i_{xz}\hat{P}_o]\hat{p} +\{[(M_{q,n}/\mu_1)+(M^{\bullet}_{w,n}/\mu_1)((Z_{q,n}/\mu_1)$$
$$+1)]\hat{q}+\{(\mu_1(i_z-i_x)/\mu^2_2)\hat{P}_o + 2(\mu_1/\mu^2_2)i_{xz}\hat{R}_o]\hat{r}$$
$$+ [(-M^{\bullet}_{w,n}/\mu_1)\hat{g}_2 \cos\Phi_o]\theta + [(-M^{\bullet}_{w,n}/\mu_1) \ \hat{g}_1 \sin\Phi_o]\Phi$$
$$+ [M_{n,n} + (M^{\bullet}_{w,n}/\mu_1) \ Z_{n,n}]\eta \tag{14}$$

$$(i_x i_z-i^2_{xz}) \ \hat{\dot{r}} = [\mu_2(i_{xz}L_{v,n}+i_x N_{v,n})]\hat{v}+[i_{xz}L_{p,n}+i_x N_{p,n}$$

$$+\hat{Q}_o(i^2_x-i_x i_y(\mu_2/\mu_1)^2+i^2_{xz})]\hat{p}+[\hat{P}_o(i^2_x -i_x i_y(\mu_2/\mu_1)^2$$
$$+i^2_{xz})+\hat{R}_o i_{xz}(i_y(\mu_2/\mu_1)^2-i_z-i_x)]\hat{q}+[i_{xz}L_{r,n}+i_x N_{r,n}$$
$$+\hat{Q}_o i_{xz}(i_y(\mu_2/\mu_1)^2-i_x-i_z)]\hat{r} + [\mu_2 i_{xz}(L_{\xi,n}\xi+L_{\zeta,n}\zeta)$$
$$+\mu_2 i_x(N_{\xi,n}\xi+N_{\zeta,n}\zeta)] \tag{15}$$

Non-dimensional orientation equations

$$\hat{\dot{\theta}} = (\cos\Phi_o)\hat{q} + (-\sin\Phi_o)\hat{r} + (-\hat{Q}_o\sin\Phi_o-\hat{R}_o\cos\Phi_o)\Phi \tag{16}$$

$$\hat{\dot{\Phi}} = \hat{p} + (\tan\theta_o\sin\Phi_o)\hat{q}+(\tan\theta_o\cos\Phi_o)\hat{r}+(\hat{Q}_o\sin\Phi_o$$
$$+\hat{R}_o\cos\Phi_o)\sec^2\theta_o \ \theta+(\hat{Q}_o\tan\theta_o\cos\Phi_o$$
$$- \hat{R}_o\tan\theta_o\sin\Phi_o)\Phi \tag{17}$$

$$\hat{\dot{\psi}} = (\sec\theta_o\sin\Phi_o)\hat{q}+(\sec\theta_o\cos\Phi_o)\hat{r}+[\tan\theta_o\sec\theta_o(\hat{Q}_o\sin\Phi_o$$
$$+\hat{R}_o\cos\Phi_o)]\theta+[\sec\theta_o(\hat{Q}_o\cos\Phi_o - \hat{R}_o\sin\Phi_o)]\Phi \tag{18}$$

where

$$\hat{g} = mg/(0.5\rho V^2 S) = C_L \sec\theta_o$$
$$\hat{g}_1 = \hat{g}\cos\theta_o = C_L$$
$$\hat{g}_2 = \hat{g}\sin\theta_o = C_L\tan\theta_o$$
$$\mu_1 = m/(0.5\rho S\bar{c})$$
$$\mu_2 = m/(0.5\rho Sb)$$

These nine linearized coupled equations are known as the state-space or state-variable equations. They can be represented mathematically as

$$\{\dot{x}\} = [A] \{x\} + [B] \{u\} \tag{19}$$
where

{x} = the state vector
{u} = the control vector
[A] = the system matrix
[B] = the control matrix

To solve the state equation (Eq. 19) numerically, it is noted that the set of output parameters [x] is continuous for finite inputs, and hence can be integrated satisfactorily by trapezoidal integration. For a set of discontinuous excitations [u], however, the best approximation proves to be by rectangular integration.[13]

The Z-transform notation is adopted for simplicity. Thus Eq. (19) may be integrated:

$$\{x\} = [A] \ I_T \ \{x\} + [B] \ I_R \ \{u\} \tag{20}$$
where

I_T = the trapezoidal integration
I_R = the rectangular integration

Taking the Z-transform of both sides of Eq. (20), and substituting the relevant transfer functions for I_T and I_R

$$\{x(Z)\} = [A] \cdot (\tau \,/\, 2) \cdot [(Z+1) \,/\, (Z-1)] \,\{x(Z)\}$$

$$+ \,[B] \cdot (\tau \,/\, (Z-1)) \cdot \{u(Z)\} \qquad [21]$$

where

$$I_T = (\tau \,/\, 2)(Z+1)/(Z-1)$$

$$I_R = \tau \,/\, (Z-1)$$

Rearrangement of Eq. (21) yields the recursion equation:

$$\{x_n\} = [I-(\tau/2)A]^{-1} \,[I+(\tau\,/2)A]\{x_{n-1}\}$$

$$+\tau[I-(\tau\,/2)A]^{-1} \,[B]\{u\} \qquad [22]$$

Equation (22) is the solution of the state equation, with a global error in the order of square of the time interval, and a local error of cube of the time interval.[14] The output of a computer program incorporating Eq. (22) provides the aircraft response following initial disturbances in the forward velocity, side velocity, downward velocity, rolling velocity, pitching velocity, yawing velocity, azimuth angle, elevation angle, bank angle, elevator deflection, rudder deflection, and aileron deflection. The program is capable of predicting the response of any parameter under any selected group of perturbations during straight or maneuvering flight.

Application

As an application for the above analysis and algorithm, the response of airplane "F" of Roskam[10] at 20 000 ft altitude is obtained under different initial disturbances and control deflections. Complete agreement is found between the stability criteria extracted from the response curves, i.e., the computer output, and the results of Roskam[10] for both the longitudinal and the lateral dynamic stability during cruising flight.

The inputs to the computer comprise the specification and type of flight (straight or maneuvering), time increment, aerodynamic and geometric characteristics, aerodynamic derivatives, disturbances, response diagrams to be illustrated, and figure scales. The program has the capability to illustrate complete response diagrams for an airplane during straight and maneuvering flight.

Computer outputs are classified into three groups of figures. The first group illustrates the linear perturbations $\hat{u}$, $\hat{v}$, and $\hat{w}$. The second and third groups present angular perturbations $\hat{p}$, $\hat{q}$, $\hat{r}$, and orientation parameters Φ, θ and ψ, respectively.

Figures 1 to 4 illustrate the responses of the aforementioned groups for an initial disturbance $\hat{u}$ of 0.05. The vertical axis represents the response in non-dimensional form while the horizontal axis represents non-dimensional time.

It is clear from these figures that the initial symmetrical disturbance on the airplane during steady straight flight produces only symmetrical perturbations, i.e., $\hat{u}$, $\hat{w}$, $\hat{q}$ and θ. This evidence may be considered as further support for the validity of splitting the equations of motion into longitudinal and lateral components for symmetrical and asymmetrical motion, respectively.

Figures 1 to 3 feature long-period, increasing oscillation modes, i.e., negatively damped phugoid modes, in the responses of forward velocity $\hat{u}$, rate of pitch $\hat{q}$, and elevation angle θ. It is evident that $\hat{u}$ and $\hat{q}$ are in-phase, while there is a phase angle of 90° between them and θ. At the beginning of time history, the rate of pitch $\hat{q}$ experiences a light overshoot followed by the negatively damped phugoid mode. The downward linear velocity response $\hat{w}$ varies very slightly, meaning that the change in the angle of attack is diminutive.
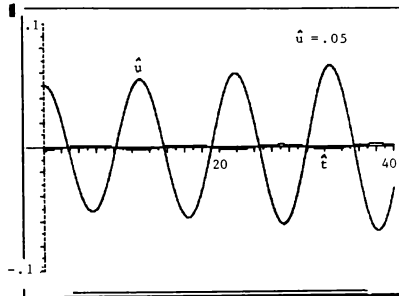

Fig. 1 Response of angular velocity to an initial disturbance of $\hat{u}$=0.05.
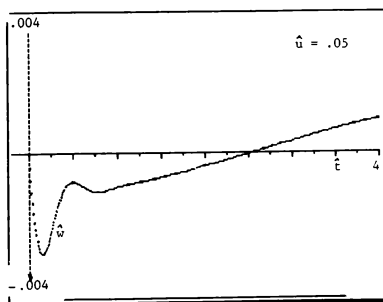

Fig. 2 Elucidation of linear disturbances for $\hat{u}$=0.05.

It may be observed in Fig. 11 that the bank angle oscillates with low amplitude about its initial value. This oscillation has the same period and phase as the yawing rate, but it is out-of-phase with the rolling rate by 180°. This figure additionally illustrates that the azimuth angle increases smoothly to a low asymptotic angle.

In summary, it is deduced from Figs. 5 to 11 that the airplane possesses the complete set of modes in an integrated form. Additionally Fig. 11 suggests the existence of a divergent spiral mode.

Figures 12 to 17 present the response of the airplane during turn maneuvering flight with a bank angle of 10°, and subjected to the following set of complicated perturbations:

1. Step function rudder deflection, zta=0.25°
2. Sideslip angle = 0.005 rad
3. Angle of attack = 0.01 rad, and
4. Rolling rate = 0.036 rad /s

At the initial stage of the time history, the forward velocity component exhibits a slightly positive rise, after which it decreases, and fluctuates with large-amplitude, long-period oscillations in the negative direction (Cf Fig. 12).
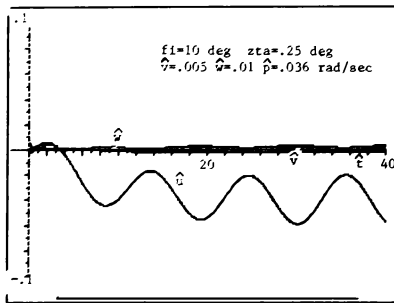

Fig. 12 Response of linear velocity to initial disturbances of 0.25° rudder deflection, $\hat{v}$=0.005, $\hat{w}$=0.01, and $\hat{p}$=0.036 during turn maneuvering flight with 10° bank angle.

Figures 13 and 14 illustrate that the sideslip angle starts its early part of motion with heavily damped, short period oscillation in the positive side, after which it decreases steadily and asymptotically to a low value in the negative side without any oscillation.

The angle of attack exhibits a highly damped subsidence through the initial stage, appearing as an overshoot. The peak of the said overshoot produces a negative change in the angle of attack. This phenomenon is followed by a low-amplitude, long-period oscillation on the positive side. Thus the angle of attack increases slightly with time. It is further observed that the long-period oscillations of the

perturbations of the angle of attack are about 180° out-of-phase with the fluctuations of the forward velocity.

The rolling rate response experiences a heavily damped rolling subsidence mode during the early stages of motion, which transfers the rolling mode to the negative direction (Cf Figs. 15 and 16). This is succeeded by a Dutch Roll mode, characterized by heavily-damped short-period oscillation. Thereafter the roll rate builds up with time, until it vanishes.
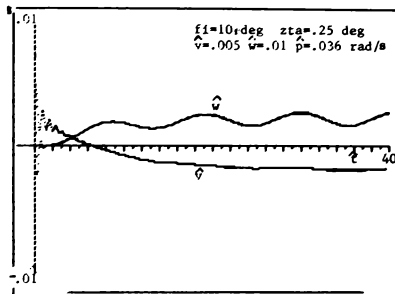

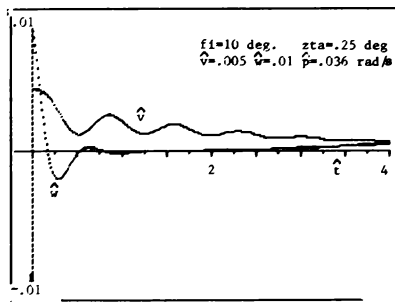Fig. 13 Elucidation of $\hat{v}$ and $\hat{w}$ of Fig. 12.

Fig. 14 Elucidation of $\hat{v}$ and $\hat{w}$ of Fig. 12.

As for the rate of yaw, there is an initial period of short-period oscillation, followed by asymptotic decay to a negative value. This phenomenon arises due to the fact that positive rudder deflections generate negative yawing moments.

It may be further observed in Figs. 15 and 16 that the rate of pitch starts with an overshoot at the initial stage of motion. The overshoot comprises a sharp deceleration to a negative value, succeeded by a rapid acceleration to a value higher than that at its initial condition. The overshoot of the yawing rate is followed by a negative deviated long-period oscillation which is in phase with the long-period oscillation of the forward velocity. The said long-period
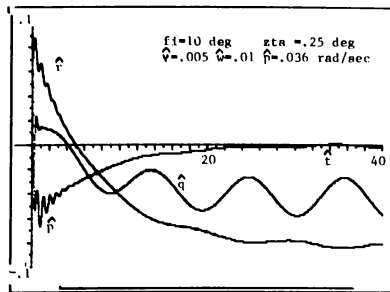
Fig. 15 Response of angular velocity to initial disturbances of 0.25° rudder deflection, $\hat{v}=0.005$, $\hat{w}=0.01$, and $\hat{p}=0.036$ during turn maneuvering flight with 10° bank angle.
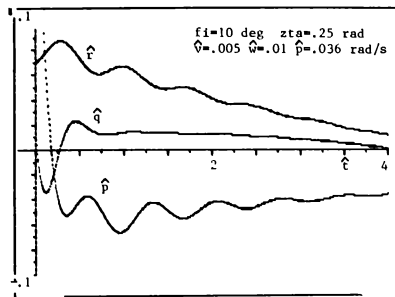

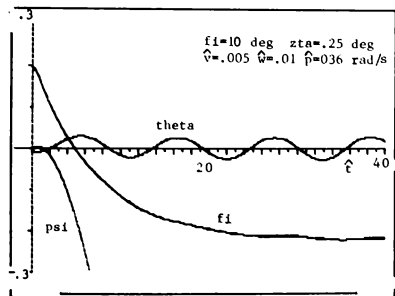**Fig. 16** Elucidation of $\hat{p}$, $\hat{q}$ and $\hat{r}$ of Fig. 15.

Fig. 17 Response of orientation to initial disturbances of 0.25° rudder deflection, $\hat{v}=0.005$, $\hat{w}=0.01$, and $\hat{p}=0.036$ during turn maneuvering flight with 10° bank angle.

mode transfers the pitching rate from the positive to the negative direction.

Figure 17 exhibits a long-period, low-amplitude oscillation for the elevation angle. Simultaneously the bank angle is seen to decrease steadily, tending asymptotically to a relatively large negative value. It is important to state that the azimuth angle increases exponentially with time on the negative side. This condition is referred to the airplane divergence mode, which becomes tighter and steeper with time.

Fig. 18 Response of linear velocity to a step aileron deflection of zai=2° during maneuvering flight with 10° bank angle, and pull-up load factor of 1.25.

Fig. 19 Elucidation of the response of $\hat{v}$ and w of Fig. 18.

Figures 18 to 22 show the response of the airplane during maneuvering flight with a bank angle Φ of 10°, a pull-up load factor lf of 1.25, and a step aileron deflection of 2°. During the initial stage of the time history, the forward velocity is accelerated exponentially with time to approximate the value of û=0.05. Thereafter it fluctuates with large amplitude and long-period oscillation with the forward velocity always positive. Figure 18 reveals that the forward velocity always positive. Figure 18 reveals that the peaks of these fluctuations increase with time.

Figures 18 and 19 demonstrate that the response of the side slip angle starts with heavily damped oscillation, followed by an asymptotic fall to a low value on the positive side. The latter value is higher than the negative angle of attack perturbation. It is further noted that the angle of attack sets on the negative side, and does not produce oscillations.
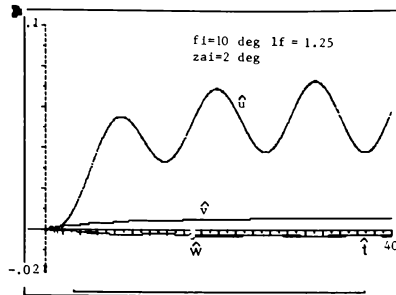

Fig. 20   Response of angular velocity to a step aileron deflection of zai=2° during maneuvering flight with 10° bank angle, and pull-up load factor of 1.25.
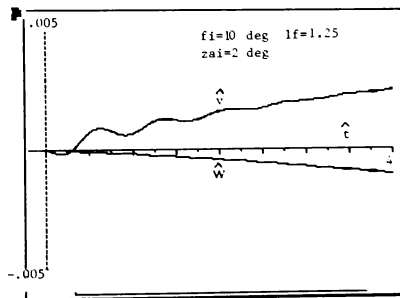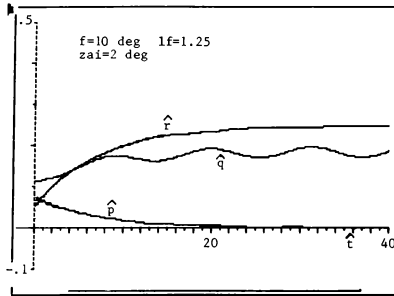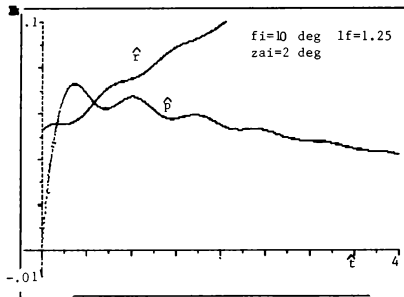

Fig. 21   Elucidation of the response of angular velocity during maneuvering flight.

Figures 20 and 21 illustrate the responses of rolling, pitching, and yawing rates. The rate of rolling exhibits a sudden rise with an overshoot followed by low-amplitude, heavily-damped oscillations of short period. Thereafter it is damped steadily with rolling subsidence mode.

The pitching rate possesses a low-amplitude, long-period oscillation that tends to increase in amplitude with time. The rate of yaw, on the other hand, increases steadily to a an asymptotic value that is higher than its initial value. This phenomenon is explained by the fact that the plane experiences a positive yawing moment derivative due to the aileron.
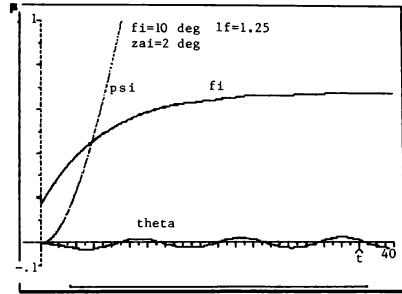

Fig. 22   Response of orientation to a step aileron deflection of zai=2° during maneuvering flight with 10° bank angle, and pull-up load factor of 1.25.

Figure 22 displays a long-period, low-amplitude oscillation for the elevation angle, while the bank angle increases steadily with time. This figure also shows the azimuth angle, which increases exponentially with time on the positive side. The latter phenomenon reflects the existence of a spiral divergence mode which becomes tighter and steeper with increasing time.

Conclusion

It is evident from the above exposition that airplane responses following initial disturbances can be predicted for maneuvering or straight flight. This is achieved by the utility of a system of first order linearized small-disturbance equations for an airplane.

Stability criteria, extracted from the response curves of the present analysis, may be observed to be in complete agreement with the results forecast by earlier workers. The foregoing conclusion seems to be valid for longitudinal as well as lateral dynamic stability during cruising flight.

References

[1]   Lanchester, F.W., "Aerodonetics", Archibald Constable, London, 1908.

[2]   McRuer, D., Ashkenas, I., and Graham, D., "Aircraft Dynamics and Automatic Control," Princeton University Press, Princeton, New Jersey, 1973.

[3]   Durand, W. F., "Aerodynamic Theory, a General Review of Progress," v 5, Peter Smith Publ., Glouchester, Mass., 1976.

[4]   Bryan, G.H. and Williams, W.F., "The Longitudinal Stability of Aerial Gliders," Proc Roy Soc Series A, 73, No 489, 100-116, 1904.

5 Bryan, G.H., "Stability and Aviation," MacMillan, London, 1911.

6 Bairstow, L., Melvill Jones, B., and Thompson, ., "Investiga tion into the Stability of an Airplane," ARC R&M77, 1913.

7 Hunsaker, J.C., "Experimental Analysis of Inherent Longitudinal Stability for a Typical Biplane," NACA TR 1, Pt.1, 1915.

8 Glauert, H., "The Longitudinal Stability of an Airplane," ARC R&M 638, 1919.

9 Glauert. H., "A Non-Dimensional Form of the Stability Equations of an Aeroplane," Br., A.R.C. R & M 1093, 1927.

10 Roskam, J., "Airplane Flight Dynamics and Automatic Flight Controls," Roskam Aviation and Engineering Corp., 1982.

11 Nelson, R.C., "Flight Stability and Automatic Control," McGraw Hill, 1989.

12 Babister, A.W., "Aircraft Dynamic Stability and Response," Pergamon Press, 1980.

13 Durling, A., "Computational Techniques: Analog, Digital and Hybrid Systems," Intext Educational Publishers, New York and London, 1974.

14 Gerald, C.F. and Wheatley, P.O., "Applied Numerical Analysis," 3rd Ed, Addison-Wesley, 1984.

A METHOD FOR DETERMINING TRANSPORT DELAYS IN THE FLIGHT SIMULATION ENVIRONMENT

by

R. Marshall Smith[*]
National Aeronautics and Space Administration
Langley Research Center
Hampton, Virginia 23665-5225

ABSTRACT

This paper will describe the process of transport delay measurement along several points in the signal path for a piloted flight simulation. Several transport delay measurement techniques were compared to determine the accuracy and ease of use of each method. Measurements were made in both the frequency and time domain. The transport delays for individual pieces of equipment as well as the math model of the simulated vehicle are measured and broken out from the system's total transport delay.

INTRODUCTION

Recent hardware and software upgrades to the Flight Simulation Facility (FSF) at NASA Langley Research Center include the addition of the Advanced Real-Time Simulation system (ARTS)[1], to provide communication links for the simulation; the Computer Generated Imagery (CGI) system, for out-of-the-window scenery; the Calligraphic/Raster Display System (CRDS), to generate displays for cockpit instrumentation; two CGI target generators, for dome simulations; new control loaders; a visual scene projection system for domes, utilizing four GE light valve projectors; and, multiple window visual systems utilizing eleven XKD monitors and five Tector monitors. The addition of these items has necessitated a reevaluation and documentation of the transport delays associated with various flight simulation hardware and software systems.

This study will document the transport delays of equipment and software associated with the Transport Systems Research Vehicle (TSRV) cockpit. This includes the TSRV control loader, the ARTS highway, the CYBER 175 real-time computer, the CGI system, the CRDS, and cockpit display monitors. In addition, the transport delay associated with the vehicle math model will also be examined.

For this study transport delay is defined as the time elapsed from a pilot input until an appropriate response is made to the pilot by the associated hardware and software (math model, motion base, CGI, CRDS, et cetera). Transport delay can be further divided into a hardware transport delay time and a software transport delay time. For the purposes of this paper the software transport delay is defined as the delay associated solely with the vehicle model that is implemented. There may also be transport delays within individual pieces of equipment that are generated by internal software (for example the CGI real-time software and databases, ARTS system communications software, et cetera). Since these delays remain constant from one vehicle model to another they are quantified as one transport delay for that piece of equipment. Therefore, the hardware transport delay is defined as any delay resulting from a piece of simulation equipment that is not associated with the math model.

The primary function of the FSF is to conduct basic aerospace research. In conducting this research daily changes are required to be made to many different math models. Also, each of these math models may require different pieces of simulation equipment. As the math model software changes, the transport delay through that software could change as a result of different programming methods or undetected programming errors. The optimum way to track transport delays in this environment would be to individually quantize the delays that are

---

[*]Engineer, member AIAA

associated with each piece of equipment and each math model. Once these delays are known then the programmer can calculate the transport delay associated with the simulation based on timing of the math model and the known transport delays of the equipment utilized. Each programmer can internally, on the simulation computer, test the math model transport delay time, which can then be added to the calculated hardware delay time. This yields a simple method for determining total transport delay in a dynamic, multi-user environment. Further testing would only be required for equipment or software that undergoes any changes.

It is well known that the transport delay must be held to a minimum in order to avoid degraded pilot performance[2]. Also, if the transport delays are too long simulator sickness can possibly occur from mismatched cueing[3]. Therefore, it is critical that one first quantify the transport delays associated with a particular simulator and, when necessary, reduce the transport delays to an acceptable range. The Federal Aviation Administration (FAA) and independent testing have confirmed that it is desirable to keep transport delays under 150 milliseconds (ms) in order to avoid degraded pilot performance[2,4]. However, there is evidence that longer delay times may be tolerated for vehicles that have lower response rates such as transport aircraft[2].

Transport delays may be measured in either the frequency domain or the time domain. During the course of this study the transport delays were measured in both the time and frequency domains and results were compared for any discrepancies. The frequency domain based measurements were conducted using a frequency response analyzer. The time based measurements were recorded utilizing a logic analyzer and a strip chart recorder. Several methods for data collection were also tested and compared. The first method utilized to collect data was the traditional photo-electric diode attached to a monitor. A second method detects video levels at the input of the monitor. Finally, a logic analyzer collected data along the digital path and yielded time based measurements. The time domain and video level detector proved to be the simplest to use while yielding accurate results and was then utilized for the remaining transport delay tests.

SIMULATOR

The TSRV simulation utilizes a math model of a Boeing 737 transport aircraft. Figure 1 shows the flight simulation hardware being tested. The simulation executes on a CYBER 175 and interfaces to the required simulation equipment through the ARTS system. The ARTS system utilizes CAMAC crates to interface hardware to the fiber optic highway. Each crate contains all the necessary Analog-to-Digital Converters (ADCs), Digital-to-Analog Converters (DACs) and digital data interfaces for a particular site. The 737 simulation requires the CGI, (an Evans & Sutherland CT6) for out-the-window visuals, one third of the CRDS, (a Terabit Eagle 1000), to generate eight Heads-Down Displays (HDD), and the TSRV cockpit. The cockpit is composed of a McFadden sidearm control loader, discrete switches and two Lear Siegler Cockpit Display Units (CDU)[5]. The CYBER 175 executes the simulation at 33 Hz or every 30 ms. The CGI updates a 771 interlaced display at 50 Hz or every 20 ms. The CRDS is a calligraphic system used mainly for HDD's and Heads-Up-Displays (HUD) but it also has raster channels that can be mixed with the CGI image to provide HUD's for simulators that do not have a HUD installed. The update rate for the CRDS is a function of the complexity of the graphical image being drawn. The more complex the image the slower the update rate. The update rate for the TSRV baseline displays is 40 Hz and for the displays used in this study the update rate is 60 Hz. Due to the different update rates throughout the system there are several points of asynchronous communication. Figure 2 illustrates all asynchronous nodes in the simulation system. These nodes can be modeled as a Zero-Order-Hold (ZOH) device with the average transport delay of one half the update rate[6].

MEASUREMENT TECHNIQUES

Photo-Electric Diode

The first method used to collect data utilized a photo-electric diode to detect changes in the ambient light emanating from a monitor. The diode method has several problems due to its design and operational characteristics. The diode should first be quantized to know its delay characteristics. Ambient light in the cockpit can interfere with measurements. The ambient cockpit light may also cause the diode to signal an event when one has not occurred. The photo-electric diode also does not take into account the interlace effect prevalent with most CGI systems. If the light level of the first field is not bright enough to trigger an output then the measurement could be off by as much as one field. Finally, data collection with a photo-electric diode is usually limited to the cockpit site.

Video Level Detector

The video level detector utilizes the incoming Red, Green and Blue (RGB) signals to detect a video level

change. Figure 3 shows the schematic for this circuit. The detector monitors the RGB signals with three comparators. Once the video level reaches a preset threshold, the comparators will output a Transistor-to-Transistor Logic (TTL) level to Schmitt-triggers which will then square the signal for output to an AND gate. The output of the AND gate will only be high when RGB are high, meaning white is detected. This signal is then fed to an Erasable Programmable Logic Device (EPLD). An EPLD is used to give a trigger output signal based on what kind of video signal is being used. The CGI system is capable of generating several different video formats. A 771 line format is used of transport simulators and an 841 line format is used for dome simulators. For each video format the RGB information modulates with each pixel being displayed. In order to conduct measurements in the frequency domain the input signal that is fed into the Frequency Response Analyzer (FRA) must have a solid high during the white field or a solid low during the black field. Standard RGB signals for a white field are not normally a solid high. The signal will have periods of lows during the horizonal retrace time and also for a period of time before, during and after the vertical retrace time. The amount of time the signal remains low depends on the line rate and number of active lines for which the video signal is configured. Because of these intermittent lows simply monitoring the RGB output is not sufficient to conduct frequency domain analysis. The optimum detector would trap and hold the state, high or low, of the first pixel of the first active line every field. This would allow the output of the detector to reflect the state of the new video field. This circuit is not necessary if one is interested in only detecting transitions from a black image to a white image in the time domain but it is necessary to detect transitions from a white to a black image or to collect data in the frequency domain. Figure 3 also shows two switches NLIM, and Raster. The NLIM switch allows different timing mechanisms in the EPLD to count different numbers of active and passive lines, depending on the format being tested and the raster switch allows the circuit to be used for calligraphic video. When calligraphic video is being tested the video input to the EPLD is routed directly to the trigger output. This will only give an output when video changes from black to white and is only useful when conducting time domain measurements.

The timing of a raster image was at a fixed frame rate. The CGI system operates at 50 Hz or 20 ms for every video field. Therefore the time for a CGI response from an input from the host computer to the completion of the first field is the time measured by the video level detector plus 20 ms. For the calligraphic systems the only useful information that can be gathered is when the graphics image has started to be drawn. The time to image completion is dependant on the time required to complete all necessary graphics.

Logic Analyzer

Finally a Hewlett-Packard 1241 logic analyzer was utilized to detect the start event, either a step input at the cockpit or digital data on the ARTS highway, and the stop event, a signal from either the photo-diode or video detect circuit.

TEST SETUP

The signal flow shown in Figure 4 depicts the entire path an input signal at the control loader must take to cause a change in the CGI or CRDS. The major components of the simulation (the control loader, the ARTS/CYBER system, the CGI and the CRDS) each have their own transport delays. The test setup for measuring the delay in each piece of equipment is described below.

Control Loader

The control loader's transport delay is measured from the input of a sinusoid or step input at node 1, the force transducer, to the output of the stick position to the host computer at node 2. First the normal inputs from the force transducer are disconnected from the signal conditioning card in the control loader so that an artificial signal may be injected in its place. A 0.0 to 2.5 Volt (V) step input and a 1 Volt peak-to-peak (V_{pp}) sinusoid signal were input in place of the force in the pitch, or θ, axis. The step input and sinusoid signals were both used to measure delay in the time domain using the logic analyzer and a strip chart recorder. The sinusoid input is also used in frequency domain timing. The measurement apparatus used for the frequency domain was a Frequency Response Analyzer (FRA). The FRA outputs a sinusoidal signal, at a particular frequency, which is then fed to the equipment to be tested. The FRA monitors the output of the equipment tested and calculates a phase difference, $\Delta\phi$, between the two signals. The phase difference can then be converted to the time domain by the following equation:

$$t \text{ (sec)} = \Delta\phi°/(360°{*}\text{frequency})$$

More detailed information on this method is contained in reference 6.

ARTS/CYBER

The ARTS/CYBER delay is measured from the input

of a step and sinusoid at the ADC, node 3, to the output of the system at the CGI crate and the Eagle crate, nodes 4 and 7. The input ADC converts the position output from the control loader into digital data suitable for the CYBER. The data is read into the CYBER at the beginning of each real-time frame (every 30 ms). For testing without the math model a statement was inserted into the real-time program that will cause the program to skip all model dependant executable statements and then output to the CGI and CRDS a signal that indicates the input ADC has detected a change at the cockpit. For testing with the math model the aircraft's pitch was monitored for any deviation from a trimmed condition to signal a response from the input. First the aircraft is placed into a trimmed condition to stabilize the pitch variable, THETADEG, or θ_{deg}. Once θ_{deg} has changed by a preset threshold amount the appropriate output is made to the CGI and CRDS indicating that the input has traveled through the math model. This is not to say that the output of the math model is correct, that requires more detailed analysis, but only that the onset of a response to a pitch input is occurring.

CGI

The CGI system was timed using the logic analyzer, detecting an input at node 4, to start the timing and the photo-electric diode or the video level detector, at nodes 6 and 5, to stop the timing. The CGI must be capable of outputting to the monitor a black and white image. This is normally accomplished by developing a small black and white database. This could result in misleading results since this is not the actual operational database. In order to test the transport delay with the operational database, a 737 aircraft was positioned in front of the eyepoint at a 90° roll angle. The eyepoint was then positioned so close to the aircraft landing gear that the entire screen was filled with a black image. By translating a few feet in altitude the screen will be filled with white from the wing of the aircraft. This method allows the detection of transition from one state to another by monitoring the incoming data for a particular pattern that corresponds to altitude. Therefore, the real-time program will send down to the CGI a preset set of coordinates (x, y, z, heading, pitch, roll) when the ADC input is low and another set of coordinates, differing only by z, when the ADC input is high.

CRDS

The CRDS was timed similar to the CGI in that the timing will use the logic analyzer, detecting an input at node 7, to start the timing and the photo-electric diode or the video level detector, at nodes 9 and 8 to

end the timing. In order to verify transport delay in the CRDS, it was necessary to develop a simple program capable of displaying a small white square. Measurements were made with the logic analyzer and the photo-electric diode or the video level circuit. The start event for timing was the arrival of the altitude information as described above. The display will toggle between black and white based on the altitude sent. The stop event will be the detection of the beginning of the image being drawn.

RESULTS

Diode versus Detector

The output of the CGI was toggled from a black to a white image by injecting a step input at the pitch ADC of the TSRV crate. There are no math model calculations being performed during this test. The output of the CGI was monitored using the photo-electric diode and the video level detector. The diode was placed as close to the upper left corner of the monitor as possible however, the collimating optics inhibited placing the probe directly on the monitor. The video level detector was placed in parallel with the video signal driving the monitor. The output from the diode, the detector and the RGB video signal were monitored by the logic analyzer. Since the RGB video signal ultimately feeds both the detector and diode it was considered the criterion by which the other signals will be measured against. Twenty measurements were taken using a step input. The average time until the video level detector transition was recorded was 114.85 ms. The average time until the photo-electric diode transition was recorded was 122.0 ms. The actual RGB video signal transition was detected at 114.84 ms. Therefore the photo-electric diode lagged behind the actual transition by 7.16 ms. Therefore, based upon the above data, the video level detect circuit was utilized for the remainder of the transport delay measurements.

Control Loader

The control loader was tested in both the frequency domain and the time domain. Since the control loader is an analog device, it begins responding almost instantaneously to a step input. Therefore, the transport delay tests that utilized are those that evaluate steady state delays due to a sinusoid input. A sinusoid was input to the control loader and the output was monitored with both the strip chart recorder and the FRA. The strip chart recorder revealed a 58 ms transport delay whereas the FRA revealed a 64.3 ms delay. The predicted transport delay is a function of the damping and the velocity feedback set at the control loader[7]. The calculated

value based on these settings was 62.6 ms. The strip chart recorder was difficult to read with high accuracy however, these results agree within the accuracy of the test setup.

Frequency versus Time Domain

A step signal and a sinusoid were input directly into the pitch ADC of the TSRV crate. Again no math model calculations were executed. The output of the CGI was monitored using the video level detector. The output of the detector was routed to the FRA while the input ADC was oscillated at various frequencies. Twenty readings were taken at each frequency. The results are shown below in Table 1.

Frequency (Hz)	Delay (ms)
0.20	114.17
0.40	116.32
0.80	118.13
1.60	117.61
3.00	117.57

TABLE 1 - FRA Results

The measurements from the frequency domain were close to those obtained in the time domain. The results were very close when the transport delays are measured using low frequencies (114.2 verses 114.9). A low frequency input is more indicative of the kind of input one would expect from a pilot during a simulation, particularly a transport simulation. Even at high frequencies the deviation is less than 4% of the total magnitude of the delay. Therefore, since the frequency and time domain measurements agreed closely with one another and much less data was required for the time domain measurements it was decided to make all remaining measurements in the time domain.

ARTS/CYBER

The step input was again input to the pitch axis ADC and the video level detector connected to the output of the CGI. The logic analyzer monitored the step input to start timing and when data arrived at the CGI. The math model was not executed for the first part of the test. Twenty measurements were made and averaged to yield the delay time. The time required for a step input in pitch to cause a change in altitude at the CGI crate was 19.6 ms. This is in agreement with the predicted value of 15 ms average to sample the ADC plus approximately 5 ms to execute the real-time program shell and to ship the required data to

the CGI. This yields a total predicted delay of approximately 20 ms. The second part of the test consisted of measuring the transport delay while the math model is being executed. Since the transport delay measured was longer than expected a detailed analysis of the software is underway to determine the source of the unexpected delay.

CGI

The CGI's transport delay was measured from the arrival of data at the CGI crate to a change in video signal level. The logic analyzer monitored the incoming data on the ARTS highway and the output of the video level detector. The CGI was tested and found to have a 92.4 ms transport delay. The predicted CGI delay was 70 ms which includes 60 ms for image generation and an average 10 ms for the CGI's Gould to sample the crate. This leaves 22.4 ms unaccounted for. Upon further investigation it was determined that the Gould front-end-computer required 20 ms to process the incoming data and 2.4 ms to retrieve the data from the CGI crate once it had been notified data was available. The CGI vendor is presently completely completing an operating system modification to remove the 20 ms delay.

CRDS

The CRDS's transport delay was measured from the arrival of data at the CRDS crate to a change in the video signal level. The logic analyzer was used in the same manner as in the CGI test. The average transport delay, for the 60 Hz update rate, measured 75.0 ms which was exactly the predicted value. The CRDS requires 4.5 frames at 16.67 ms each to compute and begin to display new data. The delay for the 40 Hz rate used by the baseline (research) displays is then 112.5 ms. The visual cue mismatch between the CRDS and the CGI (after the modification) will be approximately 20 ms which is negligible.

Total Delay

The total delay for the simulation can be calculated by adding the transport delays of individual pieces of equipment. The TSRV simulation requires all of the above pieces of equipment and therefore has a transport delay of 132 ms, to the end of the first CGI field, and 95 ms to the beginning of the CRDS image rendering (without the math model). The entire loop was measured and found to have a transport delay of 132.4 ms to the end of the first CGI field, and 94.5 ms to the beginning of the CRDS image rendering. The math model was then added to the simulation and then the overall transport delay was measured to

be the hardware times above plus the unexpected delay.

CONCLUSIONS

The video level detector provided a more accurate and reliable method to detect when data has caused a change in the output of a image or display generator. Testing conducted in the frequency domain and the time domain resulted in data that had negligible differences in the lower frequencies. Although the deviation between data was larger at higher frequencies this is not considered significant since pilot inputs are usually in the low frequency range. This is particularly true in the case of transport aircraft. The transport delays were measured for several pieces of equipment and used to calculate the overall transport delay of the simulation (with and without the math model). Through the use of this technique unexpected transport delays were discovered in the CGI and the software implementation of the math model. Once the unexpected delays are eliminated from the CGI the overall transport delay to the end of the first CGI field will be reduced to 112.4 ms without a math model. Since the transport delay through the first field of the CGI with a math model-in-the-loop is probably in excess of the FAA guidelines of 150 ms[4] efforts are being examined to accelerate the availability of CGI parameters to lower transport delays.

REFERENCES

1. Crawford, D., Cleveland, J., & Staib, R. (1988 August). The Langley Advanced Real-Time Simulation (ARTS) System. AIAA paper 88-4595, Atlanta, GA

2. Bailey, R.E., Knotts, L.H., Horowitz, S.J., & Malone, H.L. (1987, August). Effect of Time Delay on Manual Flight Control and Flying Qualities During In-Flight and Ground-Based Simulation. AIAA paper 87-2370, Monterey, CA

3. Hettinger, L.J., Lane, N.E., & Kennedy, R.S. (1988 September). Diagnostic Measurement Approaches to the Problem of Simulator Sickness in Flight Simulation. AIAA paper 88-4624-CP, Atlanta, GA

4. Federal Aviation Administration (1986 July 31). Airplane Simulator and Visual System Evaluation. Advisory Circular AC 120-40A

5. NASA Langley Research Center (1989). Langley Aerospace Test Highlights. NASA Technical Memorandum 102631

6. Johnson, W.V., & Middendorf, M.S. (1988 September). Simulator Transport Delay Measurement Using Steady States Techniques. AIAA paper 88-4619-CP, Atlanta GA

7. Parish, R.V., & Ashworth, B.R. (1976 December). The Effect of Digital Computing on the Performance of a Closed-Loop Control-Loading System. NASA TN D-8371
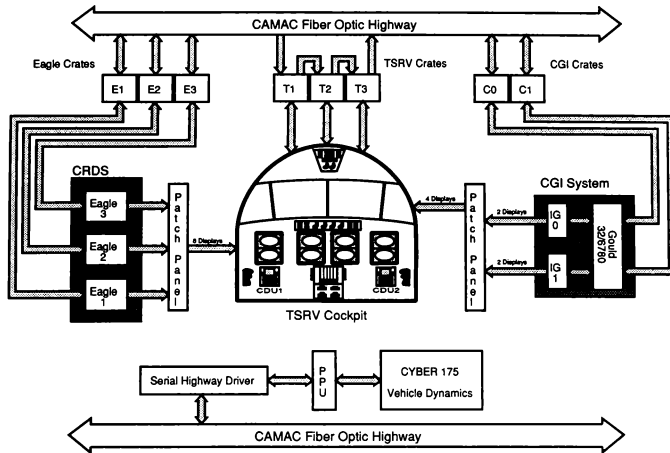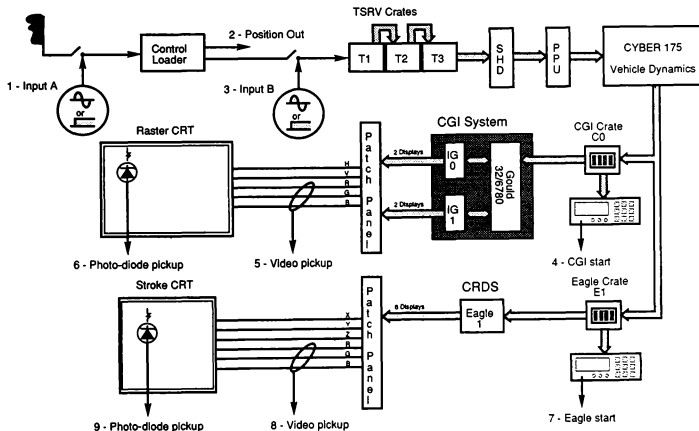
Figure 1 - TSRV Flight Simulation Hardware
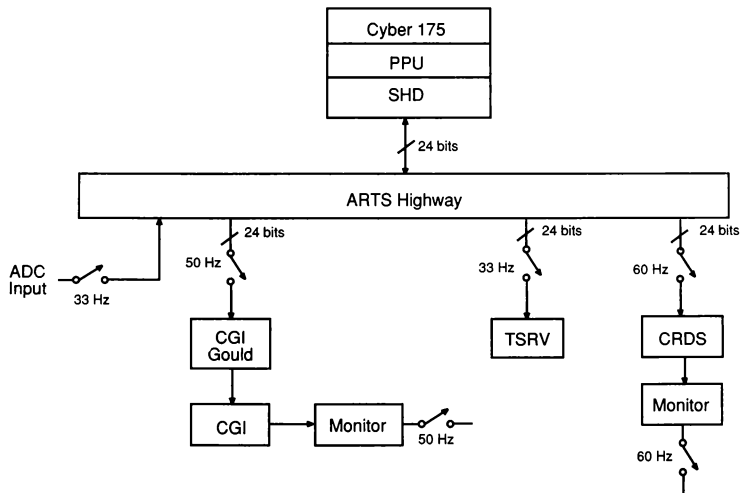
Figure 2 - Asynchronous Communication
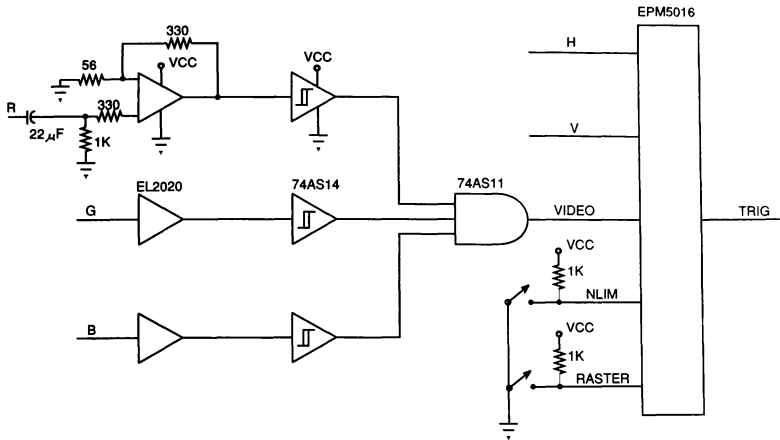
Figure 3 - Video Level Detector

Figure 4 - Signal Flow Diagram

THE EFFECTS OF SIMULATOR TRANSPORT DELAY ON
PERFORMANCE, WORKLOAD, AND CONTROL ACTIVITY
DURING LOW-LEVEL FLIGHT

Matthew S. Middendorf *
Annette L. Fiorita
Logicon Technical Services, Inc.
P.O. Box 317258
Dayton, OH 45431-7258

Grant R. McMillan
Armstrong Laboratory, CFHO
Wright Patterson Air Force Base, OH 45433-6573

Abstract

In a recent series of four experiments, test subjects were instructed to perform a low-level flight task in a fixed-base aircraft simulator with transport delays of 90ms, 200ms, and 300ms. The baseline delay condition was 90ms. Additional delay was added to the visual display loop to yield the 200ms and 300ms delay conditions.

In all statistical tests, performance was significantly degraded in the 300ms delay condition, and in some cases, it was significantly degraded in the 200ms condition. Similarly, workload significantly increased in the 300ms delay condition.

Power spectral analysis revealed a significant effect of transport delay on control activity As delay increased, subjects tended to overshoot their desired attitude (e.g., a roll to 90°), and occasionally a pilot-induced oscillation (PIO) would occur.

The addition of atmospheric disturbance had no effect on performance, workload, or control activity for this task.

The Discussion provides guidelines for acceptable simulator delay, based upon the results.

Introduction

Research on transport delay in simulators has been conducted in our laboratory for several years.[3,6,9,11,15] The purpose of the present investigation was to provide data on the effects of delay on low-level flight. These data will support the development of specifications for low-level training systems in the Aeronautical Systems Division at Wright Patterson Air Force Base.

We anticipate that low-level training systems will require high aerodynamic fidelity and scene complexity. These two factors place high computational requirements on the simulation computers which can lead to long simulator transport delay. If this delay is not properly controlled, it can have deleterious effects on simulator training effectiveness.[15] The delay manipulated in these experiments is measured from pilot input to pilot visual cuing, excluding phase shift due to vehicle dynamics. [6]

This investigation was conducted in a series of four experiments. The four experiments evolved from balancing the conflicting demands of operational realism and performance measurement. Subject motivation and incentives to discourage crashing were also a critical factor in the design of the experiments.

* Member AIAA

The displayed terrain and simulated aerodynamics were developed with several characteristics that improved operational realism. The terrain was generated to permit low level flight at a constant altitude above ground level (AGL) without exceeding the g-limits of the aircraft or the pilot. In actual flight, excessive negative G_z is avoided by pilots[2], therefore, control inputs were limited so that the aerodynamic model could generate a maximum of negative two G_z. Conversations with several active fighter pilots indicated that large angular excursions are required for low-level/terrain following flight. When performing ridge crossings pilots will typically roll the aircraft to 90 degrees and let the nose slice downward or they will completely invert the plane and pull the nose down. The aerodynamics were implemented so that both of these maneuvers could be performed realistically.

Methods

Description of the Experiments

Task Descriptions. The general task throughout the series of experiments was to fly a fighter aircraft at a low level over a rolling terrain without crashing, hitting trees, or being shot down. Four experiments were conducted in which the criterion for flying the task and other variables were manipulated (Table 1). All four of the experiments were conducted in the same manner. Each trial began when the subject informed the experimenter that he was ready. At that point, the aircraft was in straight and level flight at a speed of 480 knots and an initial altitude of 200 feet above the terrain for Experiment One, and 175 feet for Experiments Two, Three, and Four. The subject had 15 seconds to fly the aircraft to the desired altitude before scoring began. The scoring interval lasted 102 seconds. Unless they crashed, feedback to the subjects was displayed on the screen following each trial (Table 1). Delays were presented in counter-balanced blocks of trials. Refer to the Experimental Design section and Table 3.

In **Experiment One**, the subjects were instructed to fly the aircraft consistently as low as they felt comfortable within a 200 foot zone that ranged from 100 feet to 300 feet. The performance criteria in this experiment were an attempt to balance operational realism and performance measurement. Specifically, current F-16 pilots indicated that each pilot tends to have an individual comfort zone, dependent on their skill level. By allowing the

Table 1. Task Characteristics.

Exp #	Task	Disturbance	Feedback	Penalty for crashing/shot down	Incentive
1	Fly as low as comfortable within a 200 foot zone. (100' to 300')	None	SD of altitude (AGL)	No feedback	Compete for list of top 20 scores
2	Fly at a fixed altitude of 175feet	1/2 of the trials	Mean , SD, and RMS of altitude error	No feedback, Score not eligible for list, average score penalized, and subject number posted on crash list	Compete for lists of top ten RMS scores and top four average RMS scores
3	Fly as low as possible without crashing	All trials	Mean altitude (AGL)	No feedback, average score penalized, and subject number posted on crash list	Compete for list of top four average scores
4	Fly as low as possible without crashing	All trials	Mean altitude (AGL)	Termination from the experiment	Monetary award, no posted list

subjects to select a comfortable altitude within the zone, and measuring how consistently they maintained it, we attempted to allow for individual differences and still obtain stable performance measures. In this experiment, a trial was considered a crash trial only if the aircraft made contact with the ground. The subjects were instructed to avoid hitting trees, but were not penalized if they did. The standard deviation of altitude AGL was displayed after each trial as feedback. A list containing the best 20 scores was posted after 90 trials to generate competition and improve scores.

In **Experiment Two**, the subjects were instructed to fly the aircraft at an assigned altitude of 175 feet AGL. There were three reasons for the selection of this criterion: 1) to further reduce between-subject variance, 2) to prevent subjects from flying through the trees, and 3) to decrease the crash rate. Turbulence was also added for half of the trials in an attempt to increase the control bandwidth of the task. Since it is unrealistic to fly through the trees, a trial was considered a crash if the subject flew below tree top level (i.e., below 60 feet AGL). In addition, if the subject flew above 300 feet AGL for more than seven seconds, he was considered shot down by enemy fire, and the trial would be treated as a crash.

In Experiment Two, the mean, standard deviation (SD), and root-mean-square (RMS) of altitude error were displayed after each non-crash trial as feedback. The subjects understood that RMS was the criterion variable. Six posted lists were available for subject competition. Three of the lists contained the best ten RMS scores per delay condition. The other three lists contained the four best average scores per condition. These lists were intended to encourage consistent performance. The average was computed from the last 5 trials of each block of eight. The first three trials of each block were not used in calculating the average score to prevent order effects (i.e., switching between delay conditions) from unfairly inflating a subject's average score. The purpose of the lists was to provide motivation through competition. The reason we used two types of lists (single scores and average scores) was to encourage good performance and consistent performance.

In Experiment Two, crashing was strongly discouraged. If a subject crashed, his score was not displayed as feedback, that score was not eligible for the list of low individual scores, his average score was weighted, and his subject number was added to a crash list. The average score was weighted according to the following equation: posted average = real average of non-crash trials x (5 ÷ # of non-crash trials).

In **Experiment Three**, the subjects were instructed to fly the aircraft as low as possible without crashing. This task was chosen to see if delay would have a greater effect on mean altitude when a fixed reference was not provided. Specifically, if the delay was troublesome to the subjects, they might fly higher to avoid crashing. Turbulence was included for all the trials. Mean altitude AGL was displayed to the subjects as feedback following each non-crash trial. The trial was flagged as a crash if the subjects flew below tree top level, 60 feet AGL, or if they flew above 300 feet AGL for more than seven seconds. This experiment included the same penalties for crashing as Experiment Two. To further discourage crashing, a list of the best individual trial scores was not used. Only three lists containing the four most consistent performers, per condition, were used. Therefore, if a subject crashed it would decrease his chance to get on a top performers list.

Experiment Four was similar to Experiment Three with the exception of incentive and penalty for crashing. In this experiment, if the subject crashed, his participation in the experiment was terminated. If he didn't crash, he had the opportunity to earn monetary awards for good performance. There were four awards available per delay condition. First place earned $15.00; second, $10.00; third, $7.50; and fourth, $5.00. Lists of the best scores were not posted in this experiment. The purpose of this experiment was to determine the effect of strong incentives on low-level flying behavior. It was hypothesized that as delay increased, the subjects would fly higher to avoid crashing. To provide an incentive to keep them performing well, the monetary awards were available. The

subjects were eligible for an award only if they completed a delay condition without crashing. Once they crashed, they were no longer able to participate and they were ineligible for an award for that condition. However, they retained awards earned in any previously completed delay conditions. (See Table 1 for a summary of the experiments).

Workload Metric. The NASA Task Load Index (TLX), developed by the Human Performance Research Group at NASA Ames Research Center[12], was used in the evaluation of workload for all four experiments. TLX is a multi dimensional rating procedure that provides an overall workload score based on a weighted average of ratings on the following six subscales:

1) **Mental Demand** - How much mental and perceptual activity was required.
2) **Physical Demand** - How much physical activity was required.
3) **Temporal Demand** - How much time pressure was felt due to the rate or pace at which tasks or task elements occurred.
4) **Performance** - How successful the subject felt he was in accomplishing the goals of the task.
5) **Effort** - How hard the subject had to work to accomplish his level of performance.
6) **Frustration** Level - How insecure, discouraged, irritated, stressed and annoyed the subject felt.

We chose the TLX metric because it is easy for the subjects to make accurate discriminations, the individual scales provide interesting diagnostic information, and it has excellent test-retest reliability.[16]

Experimental Design. Data collection was accomplished for all four experiments using a within-subject design. Multiple 3X3 Latin squares were used to balance the order of presentation of the 90ms, 200ms, and 300ms time delays. In Experiment Two, the disturbance/no-disturbance condition was counterbalanced by an ABBA type ordering of the Latin squares. Task Load Index (TLX) workload questionnaires were also administered throughout the experiment. The number of subjects, trials, etc. is summarized in Table 3.

Subjects

Eight college-age male volunteers were paid to participate in Experiment One. All were right handed and had normal or corrected-to-normal vision. All had previous experience with flight simulators, but were naive to the low-level flight task. Six of the eight subjects returned for Experiments Two, Three, and Four.

Apparatus

Simulated Aircraft. The experiments were conducted in a fixed-base simulator with simplified fighter-type dynamics. The dynamics were implemented on a Digital Equipment Corporation PDP 11/60 using transfer functions. The roll-rate dynamics were first order with the break frequency at 6.0 radians per second. The pitch-rate dynamics were first order over second order with the zero at 1.68 radians per second, the pole (short period) at 3.84 radians per second, and short period damping ratio (zeta) of 0.8. Angle of attack was generated by passing the resulting aircraft pitch rate through a first order filter with a break frequency at 1.68 radians per second. The transfer function coefficients were linearly approximated from a full non-linear, six degree-of-freedom simulation of the the F-16 C/D block 40 aircraft flying at 480 knots at an altitude of 200 feet.[1] The discrete implementation of these filters was accomplished using the Tustin, bi-linear, Z-transform method. Integrations were accomplished using the trapezoidal rule.

The simulated aircraft was controlled using a side-mounted isometric force stick. A breakout force of 2.2 newtons was used for both axes. The roll rate scaling was 4.5 degrees per second per newton (20^0/sec/lb). The pitch rate scaling was 0.32 degrees per second per newton (1.4^0/sec/lb). These units of force assume a moment arm of 3.75 inches (0.096m) from the base of the stick.

For the sake of operational realism, the calculation of vehicle yaw rate was modified

from a simple coordinated turn approach to an enhanced yaw rate of 10 degrees per second when the aircraft was rolled to 90 degrees. The value of 10 degrees per second was subjectively determined through test trials using active fighter pilots. This enhanced yaw rate causes the nose to drop more rapidly (commonly referred to as a slice) when performing ridge crossings. The slicing of the nose is an aerodynamic effect[10] that can be increased through the use of rudders. This simulator did not have rudders, so the value of 10 degrees per second represents an average value pilots would use when presented with the simulated terrain.

Display. The display was an out-the-window scene generated by a Silicon Graphics Iris 2400T. The scene consisted of rolling terrain with a tree line down the center. The scene was projected onto a flat, matte-white screen by an Electrohome ECP graphics projector. The scene was 6 ft(1.84 m) high by 9.33 ft (2.86 m) wide with a corresponding field of view of 59.5 degrees vertical by 83.3 degrees horizontal. Subjects were seated 5.25 ft (1.61 m) from the screen.

The terrain elevation varied only in the longitudinal axis. The variation was controlled using a sum-of-sine-waves approach. The parameters used in the sum-of-sines are listed in Table 2. The initial phase for each sine wave was randomized so that the terrain was different for each trial to prevent the subjects from learning its shape.

The surface of the terrain was shaded using a palette of browns, ranging from light tan to dark brown. The shade of each polygon was selected as a function of its slope (e.g., the

Table 2. Values used in the Sum-of-Sines Generation of the Rolling Terrain.

Period (Feet)	Amplitude (feet)
7920	575
10,560	425
15,840	375
21120	250

steeper the slope the darker the brown). This shading technique is used in many other simulators[13] and is representative of the lambertian diffuse reflector.[5] To use the scene generator most effectively, the emphasis was placed on maximizing tree density rather than implementing object detail (e.g., texture).[7] The trees were 80 feet tall for Experiment One and 60 feet tall for Experiments Two, Three, and Four. They were concentrated near the center of the display using a normal distribution. The trees were slightly perturbed in the longitudinal axis using a uniform random process.

The display included a moving bar altimeter that indicated altitude above ground level (AGL). The altimeter bar was color coded according to three zones. The bar was red between 0 and 100 feet AGL, yellow between 100 and 300 feet, and blue from 300 to 500 feet. The altimeter included a scale on the right side with tick marks in increments of 100 feet. For the second experiment there was a smaller reference mark at 175 feet. The altimeter was located 14 degrees to the right of centerline similar to the altimeter on the F-16 HUD.

Disturbance. A low-altitude atmospheric turbulence model was implemented for Experiments Two through Four. This model was based on the Dryden form and had a vertical gust RMS amplitude (σ_w) of 7.7 feet per second.[4] The vertical gust was passed through spectral shaping filters to produce perturbations to aircraft pitch rate, roll rate, and angle of attack.

Delay Verification. To measure the simulator transport delay, several sinusoidal test frequencies were substituted for stick inputs. A photocell was used to measure the simulator response on the visual display. The phase shift between the input and the output was determined using a frequency response analyzer (Bafco model 916). The phase shift due to the aircraft dynamics was subtracted from the measured value at each of the test frequencies. The remaining phase shift and the input frequency were used to calculate the simulator transport delay. The minimum transport delay that we were able to achieve was 90ms (the baseline condition).[6]

Procedures

Familiarization. Experiment One included a demonstration of the basic flight maneuvers (e.g., slicing, inverting, etc.). The subjects were then permitted to fly the simulator for ten minutes without scoring. The subjects next performed 30 familiarization trials (10 per delay condition). In Experiments Two and Three, training trials were included to allow the subjects to learn the new task, if necessary. See Table 3.

Data Collection was conducted in sessions. In Experiment One, twenty trials were performed per visit and workload ratings were collected every five trials. In Experiments Two, Three, and Four, 24 trials were performed per session and workload ratings were collected every 8 trials. See Table 3.

The collection of subjective workload data was accomplished in two steps. First, at the conclusion of each set of trials, the subject would assign a value ranging from low to high to each of the six factors. Second, at the conclusion of each experiment, the subject evaluated the contribution of each factor (its weight) with respect to the task used in the experiment.

Debriefings. The design of these experiments included manipulations of task criteria, scoring, feedback, etc. To assess the impact of the manipulations, questions were prepared for post-experiment interviews with the subjects. Each questionnaire ranged from approximately 10 to 30 questions in length and was generated to be experiment specific. Another purpose of the questions was to address issues raised during the experimental design discussions. For example, did posting a crash list affect subject motivation? Similar questions are answered in the Results section. By giving the subjects open-ended, written questions and then verbal debriefings, we learned a great deal of information about the experiment. Subject feedback actually helped us improve the design of each successive experiment.

Data Analysis. Three separate dependent measures were analyzed, including performance data, workload data, and control activity data. The performance metric that was provided to the subject on the feedback display (i.e., the variable they were trying to optimize) was the variable used in the performance analysis.

In Experiment One, subjects were instructed to minimize their variance about an

Table 3. Data Collection Parameters.

Exp #	# of Subjects	Training/ Familiarization Trials	Performance Evaluation Trials	Trials per Block	TLX
1	8	1) Demonstration of the basic flight maneuvers 2) Practice for ten minutes - no scoring 3) 30 familiarization trials	120	10	Every 5 trials
2	6	24 with disturbance 24 without disturbance	24 with disturbance 24 without disturbance	8	Every 8 trials
3	6	24 with disturbance	24 with disturbance	8	Every 8 trials
4	6	6 with disturbance (2 per delay condition)	24 with disturbance	8	Every 8 trials

individually-chosen altitude AGL. Therefore, standard deviation of altitude AGL was the performance measure analyzed. In Experiment Two, subjects were instructed to maintain a fixed altitude AGL of 175 feet. Therefore, the RMS of altitude error was used in the analysis. In Experiments Three and Four, subjects were instructed to fly as low as possible without crashing. Therefore, mean altitude AGL was used. Over the course of the four experiments an increasing emphasis was placed on not crashing. However, only in the final three experiments were specific penalties employed to discourage crashing. Therefore, the performance analysis also includes crash rates, but only for the last three experiments.

In Experiment One, the first 30 trials (10 per condition) were considered as familiarization and were not included in the analysis. Despite this, subject learning was apparent over the remaining trials. Therefore the data were log transformed and a linear regression was fit to the data for each subject in each delay condition. This regression was used to predict the subjects' final (asymptotic) performance for a given delay condition. These predicted endpoints were then submitted to an analysis of variance (ANOVA). As shown in Table 3 training phases were included for Experiments Two and Three to allow subjects to learn the new task, including flying with turbulence. However, little or no learning was evident in these data. Therefore, we did not use the log transform and linear regression to reduce these data. For these experiments, means were computed using all trials for each subject under each delay condition. These means were then submitted to an ANOVA. In all experiments crash trials were eliminated from the analysis.

The analysis of the TLX workload data was the same for all four experiments. The subject weightings that were collected after each experiment were applied to the workload ratings that were collected during the experiment to yield an overall measure of subject workload for each subject in each delay condition. These overall measures were submitted to an ANOVA. The six scales used in the workload index were individually analyzed as a test of internal validity.

Power spectral analysis was used to investigate the effects of transport delay on control activity. First, all trials were processed by an FFT to generate Fourier coefficients. The magnitudes (i.e., vector sum) of the coefficients were written to power spectra files. For Experiment One these files were then averaged by subject for each block of ten trials. Since delay was constant for each block of ten trials and there were 50 trials per delay condition, this resulted in five power spectra per subject per delay condition. Each group of five power spectra was inspected to ensure that averaging across them was reasonable. Specifically, we did not want to average out a consistent trend if one existed (e.g., reduced power across blocks due to the subjects learning to compensate for delay). A consistent trend did not appear and the five power spectra for each delay were averaged to yield three power spectra per subject, one per delay condition.

A consistent effect of delay was observed in the averaged power spectra (Figure 1). There was a systematic increase in the lateral and longitudinal stick power in the 0.6 Hz and 1.9 Hz regions as delay increased. The median power in a frequency band from 0.5 to 0.7 Hz and a second band from 1.8 to 2.0 Hz were used in an ANOVA.

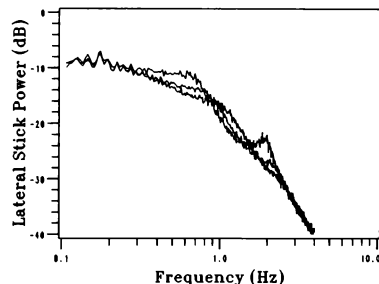Time history plots of stick activity were examined to determine the cause of the

Figure 1. Average Power Spectra.

systematic increase in power. The spectral content of the trial chosen for the time history investigation was representative of the overall average power spectrum. Specifically, we chose the subject average that best matched the overall average, then the block average that best matched the subject average, and finally, the trial within the block that best matched the block average. A second representative trial from a different subject was also examined.

The power spectral analysis for Experiment Two was similar to the procedure used for Experiment One except for the presence of the disturbance/no-disturbance condition. The two disturbance conditions were averaged separately and compared. A difference due to disturbance was not apparent and the two were averaged together. The power spectral analysis for Experiment Three was similar to the procedure used for Experiment Two except that the disturbance was always present.

Results

Experiment One

The analysis of performance data (standard deviation of altitude AGL) from Experiment One showed a significant effect of delay ($F(2,14) = 9.74$, $p < 0.0022$). Pairwise Tukey HSD comparisons showed that the standard deviation of altitude increased between the 90ms and 300ms delay conditions, and between the 200ms and 300ms delay conditions. The 90ms delay condition was not different from the 200ms condition (See Figure 2). A similar result ($F(2,14) = 10.32$, $p < 0.0018$) was found for mean altitude AGL. Although this was not the criterion variable, it was interesting to note that subjects flew significantly higher under the 300ms condition.

The effect of delay on workload was significant ($F(2,14) = 8.38$, $p < 0.004$). Pairwise Tukey comparisons showed an increase in workload between the 90ms and 300ms conditions. The other pairwise comparisons were not significant

The increase in lateral stick power due to increased transport delay was significant in the
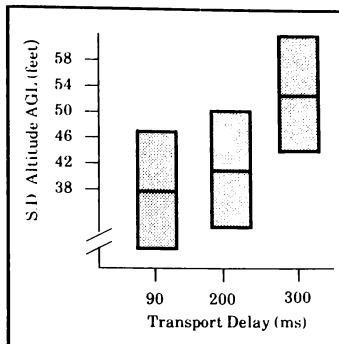
Figure 2. Experiment One: Performance Data with Tukey's 95% Confidence Intervals

0.6 Hz region ($F(2,14) = 46.35$, $p < 0.0001$). The increase in power in the 1.9 Hz region was also significant ($F(2,14) = 50.60$, $p < 0.0001$). Pairwise comparisons showed for the 0.6Hz data that all three delay conditions were significantly different from each other (Figure 3). In the 1.9 Hz region, the 300ms delay condition was significantly different from the other two, but 90ms was not different from 200ms (Figure 4). To determine why power significantly increased in these two frequency regions, time history data were examined.
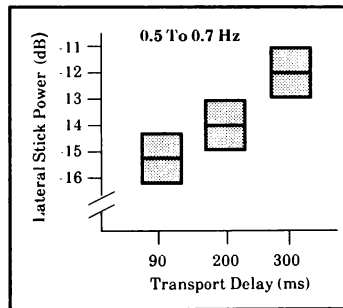

Figure 3. Experiment One: Power Spectral Data with Tukey's 95% Confidence Intervals
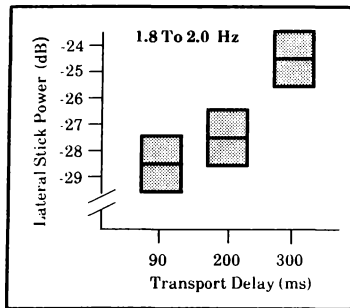
Figure 4. Experiment One: Power Spectral Data with Tukey's 95% Confidence Intervals

A time history plot (Figure 5) of lateral stick activity showed that as the subject rolled the aircraft to 90 degrees to slice the nose downward, there was considerable overshoot, and for a short time, a pilot-induced-oscillation (PIO) was present. This control activity is near 0.6 Hz. The overshoot due to increased time delay is consistent with control theory which states that as time delay in a closed-loop system increases, the damping of the system decreases.[8] The reduced damping results in the closed-loop man-machine system becoming less stable under the higher delay conditions. This effect of delay on control activity is consistent with previous research conducted in our laboratory in which subjects performed a side-step landing maneuver under the same three delay conditions.[11]

The cause of the increased power in the 1.9 Hz region was not as easy to isolate as the 0.6 Hz power. However, careful examination of time history data showed that the 1.9 Hz
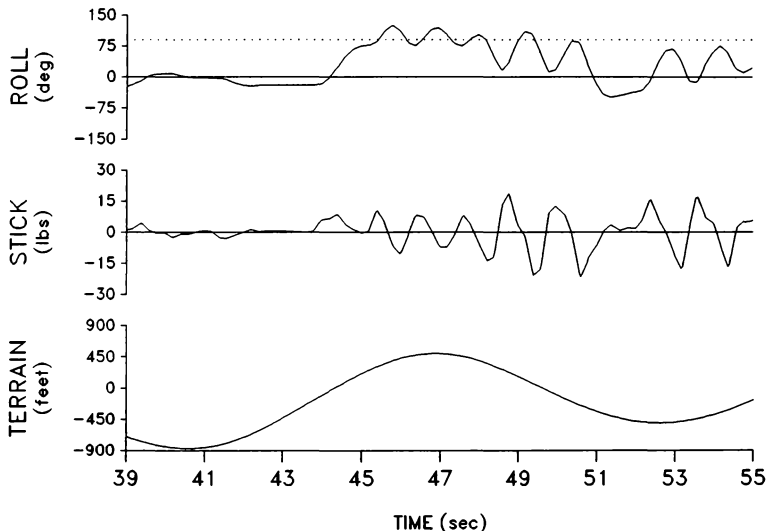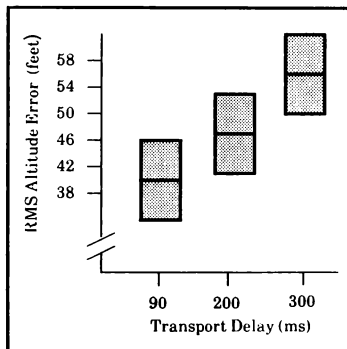

Figure 5. Time History Data From a 300ms Trial Illustrating a PIO

component usually appeared at the end of a trial and briefly following a PIO. It is reasonable to believe that this component is a neural-muscular resonance due to fatigue.[14]

Lateral stick activity was used in all of the power spectral analyses, however, peaks in the 0.6 Hz and 1.9 Hz regions also appeared in the longitudinal stick activity. Careful examination of time history plots showed that the peaks in the longitudinal stick were the result of cross-coupling. First, the points of inflection for the two signals occurred exactly at the same point in time. Second, examination of terrain shape and the subject's initial input showed that the subject would first attempt to roll the aircraft to 90 degrees and then overshoot leading to a PIO which cross-coupled into the longitudinal axis.

Experiment Two

The effect of delay on performance (RMS altitude error) for Experiment Two was significant ($F(2,10) = 28.96$, $p < 0.0001$). Altitude error significantly increased with delay for all pairwise tests (Figure 6). The effect of the disturbance/no-disturbance condition was not significant ($p > 0.05$). In fact, performance with and without the disturbance was nearly identical.
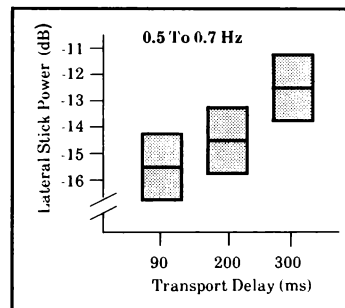
The effect of delay on workload was significant for this experiment also ($F(2,10) = 5.22$, $p < 0.028$). Pairwise comparisons showed a significant increase in workload only between 90ms and 300ms. Of the individual scales, the differences in performance and frustration were significant ($F(2,10) = 5.88$, $p < 0.0205$) and ($F(2,10) = 8.6$, $p < 0.0067$) respectively. All six of the scales showed the same trend as the overall TLX measure.

The power spectral analysis of lateral stick activity showed a significant effect of delay ($F(2,10) = 16.71$, $p < 0.0006$) in the 0.6 Hz and 1.9 Hz regions ($F(2,10) = 18.04$, $p < 0.0005$). Pairwise comparisons showed a significant increase in power between 90ms and 300ms and between 200ms and 300ms in both regions. (Figures 7&8).

The overall crash rates for Experiment Two were 0.5%, 3.1%, and 6.2% for the 90, 200, and 300ms delay conditions, respectively.

Experiment Three

The effect of delay on performance (mean altitude AGL) for Experiment Three was significant ($F(2,10) = 17.78$, $P < 0.0005$). Subjects flew significantly higher under the 300ms delay condition than under the other two, but, 90ms and 200ms were not different

Figure 6. Experiment Two:
Performance Data with Tukey's
95% Confidence Intervals

Figure 7. Experiment Two:
Power Spectral Data with Tukey's
95% Confidence Intervals

Figure 8. Experiment Two:
Power Spectral Data with Tukey's
95% Confidence Intervals
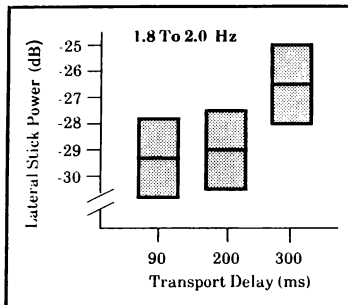
Figure 9. Experiment Three:
Performance Data with Tukey's
95% Confidence Intervals

(Figure 9). Not only did the subjects fly higher when delay was increased, but they also had greater variability. The standard deviations for the 90, 200, and 300ms delay conditions were 34, 39, and 50 feet, respectively.

Analysis of the workload data showed that the effect of delay was significant $(F(2,10) = 5.361, p = .026)$. Pairwise comparisons showed a significant increase in workload only between 90ms and 300ms. Of the individual scales only effort was significant $(F(2,10) = 5.88, p < 0.0205)$. All six scales showed a trend consistent with the overall TLX measure.

The power spectral analysis of lateral stick activity for Experiment Three showed a significant effect of delay $(F(2,10) = 10.46, p < 0.0035)$ in the 0.6 Hz and 1.9 Hz regions $(F(2,10) = 9.48, p < 0.0049)$. Pairwise comparisons for 0.6 Hz showed that the 90ms delay condition had significantly lower power than 200ms and 300ms. The 200ms and 300ms conditions were not different (Figure 10). The pairwise comparisons at 1.9 Hz indicated that only the 90ms and 300ms conditions were different (Figure 11).

The crash rates for Experiment Three were 5.9%, 6.3%, and 9% for the 90, 200, and 300ms delay conditions, respectively.

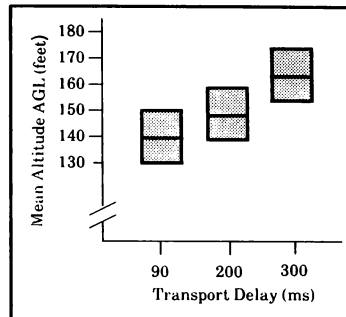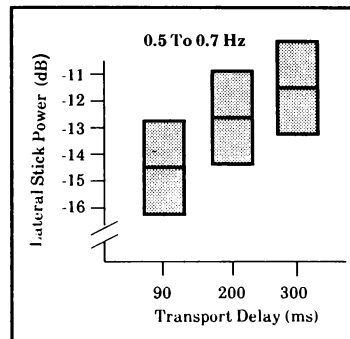Figure 10. Experiment Three:
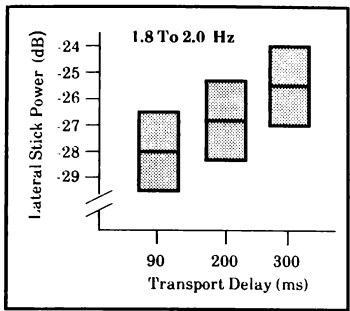Power Spectral Data with Tukey's
95% Confidence Intervals

Experiment Four

The primary purpose of Experiment Four was to determine if strong incentives would provide enough motivation to prevent subjects from crashing. In addition to the monetary awards, the subjects were also terminated from the experiment if they crashed. Two of the six subjects crashed in the first block of trials. A

third subject crashed in the third block of trials and a fourth subject was unavailable for participation. Only two subjects completed all 24 trials.

Due to the limited amount of data, statistical analysis could not be meaningfully performed. However, the results suggest that providing a monetary award did not provide

enough additional incentive to eliminate crashing. Two of the subjects indicated during the debriefings that being terminated from the experiment was a stronger motivater than the monetary awards.

Although only two subjects completed Experiment Four without crashing, their performance data are noteworthy. They consistently flew higher to avoid crashing (Table 4). For these two subjects it appears that the fear of being terminated, and possibly losing the opportunity to earn monetary awards, caused them to fly in a more realistic manner.

Figure 11. Experiment three: Power Spectral Data with Tukey's 95% Confidence Intervals

Table 4. Comparison of Mean Altitudes from Experiments 3 and 4.

	Sub. #1		Sub. #2	
	Exp 3	Exp 4	Exp 3	Exp 4
90ms	113.5	135.5	144.2	160.1
200ms	126.5	148.4	151.3	166.8
300ms	157.7	157.9	144.0	176.7

Table 5. A Summary of the Effects of Simulator Transport Delay on Performannce and Workload.

Experiment Number	Performance Analysis		Workload Analysis	
	Significant Effect?	Pair-Wise Differences	Significant Effect?	Pair-Wise Differences
1	Yes $F(2,14)=9.74$, $p<0.0022$	90 VS. 300ms 200 VS. 300ms	Yes $F(2,14)=8.38$, $p<0.004$	90 VS. 300ms
2	Yes $F(2,10)=28.96$, $p<0.0001$	90 VS. 200ms 90 VS. 300ms 200 VS. 300ms	Yes $F(2,10)=5.22$, $p<0.028$	90 VS. 300ms
3	Yes $F(2,10)=17.78$, $P<0.0005$	90 VS. 300ms 200 VS. 300ms	Yes $F(2,10)=5.36$, $p<0.026$	90 VS. 300ms
4	Insufficient Data	Insufficient Data	Insufficient Data	Insufficient Data

Table 6. A Summary of the Effects of Simulator Transport Delay on Control Activity using Power Spectral Analysis.

| Experiment Number | Power Spectral Analysis of Lateral Stick Activity | | | |
| | 0.5 to 0.7 Hz | | 1.8 to 2.0 Hz | |
	Significant Effect?	Pair-Wise Differences	Significant Effect?	Pair-Wise Differences
1	Yes $F(2,14) = 46.35$, $p < 0.0001$	90 VS. 200ms 90 VS. 300ms 200 VS. 300ms	Yes $F(2,14) = 50.60$, $p < 0.0001$	90 VS. 300ms 200 VS. 300ms
2	Yes $F(2,10) = 16.71$, $p < 0.0006$	90 VS. 300ms 200 VS. 300ms	Yes $F(2,10) = 18.04$, $p < 0.0005$	90 VS. 300ms 200 VS. 300ms
3	Yes $F(2,10) = 10.46$, $p < 0.0035$	90 VS. 200ms 90 VS. 300ms	Yes $F(2,10) = 9.48$, $p < 0.0049$	90 VS. 300ms
4	Insufficient Data	Insufficient Data	Insufficient Data	Insufficient Data

Debriefings

The debriefing questionnaires given to the subjects following each experiment yielded some interesting results. The subjects ranked the order of difficulty of the three tasks from flying within a zone as the easiest, followed by flying at a fixed altitude, and finally flying as low as possible as the most challenging. Interestingly though, subjects preferred flying as low as possible. Turbulence and the possibility of being shot down were not factors in the experiments. The subjects reported that they always knew when they were flying under high delay (300ms), but could not clearly distinguish between the two lower delays (90 vs. 200ms). They reported always trying to do their best, even under high delay. Posting lists of the best scores was very motivating to the subjects. They all reported trying to continually improve their performance in order to get their scores on the lists. The posted crash list was very effective. Subjects did not want to be on it.

Discussion

In Experiment Two, the addition of a strong atmospheric disturbance had little or no effect on performance. The most prudent explanation is that the large angular excursions required to perform the task far exceed the angular perturbations caused by the disturbance. During the post session interviews, one subject remarked that he didn't even realize the disturbance was active. One of the fighter pilots who evaluated the low-level disturbance model commented that without motion cuing the disturbance would have no effect. This assertion seems quite reasonable when examining pilot cuing via the visual system and the motion system due to disturbance. For example, a small but sudden vertical perturbation may not be noticeable in the displayed scene, however, motion cues, if provided, could be very salient.

In an attempt to balance the demands of operational realism and sensitive performance measurement, four low-level experiments were conducted. While each of the four allowed us to observe different aspects of performance, the

results show excellent agreement. Changing the task criteria, scoring, etc., did not substantially change the effects of transport delay. None of the experimental designs, including the monetary reward structure, reduced the crash rates to realistic levels. The lowest rates were observed in Experiment Two, where a fixed altitude was assigned.

The four experiments clearly indicate that simulator transport delay degrades a pilot's ability to perform accurate low-level flight. This conclusion is supported by the performance, workload, and stick activity data. Applying these results to an operational simulator requires consideration of two important caveats. First, the current experiments only considered the terrain-following aspects of low-level flight. No explicit lateral maneuvering or terrain-avoidance was required. Addition of this dimension may increase the effects of delay. Second, the aircraft dynamics implemented in this simulation used linear transfer functions. The simulated dynamics did not include actuator lags, flight control computer delays, or other high-order dynamics that produce additional delay in the actual fighter. In the F-16, for example, these components add between 80 and 100ms of equivalent delay to the bare airframe dynamics. Given this fact, our baseline delay case (linear dynamics plus a 90ms transport delay) should be considered representative of a modern fighter aircraft.

From this perspective, adding 210ms of delay to the baseline case leads to an unacceptable simulation. A simulator with delays in this range will require the pilot to fly higher to avoid crashing, will increase altitude variability, will lead to increased stick activity, and will increase pilot workload. Despite this degradation of in-simulator performance, delays in this range may not eliminate the training benefit of such a simulation. Other delay research conducted in this laboratory suggests that effective training can occur despite the performance degradation.[15]

The results of the four experiments do not clearly indicate if adding 110ms of delay to the baseline case is unacceptable. While most of the statistical tests found no difference between

the 90 and 200ms delays, stick activity was affected in two of the experiments, and performance was degraded in Experiment Two. Previous experience in this laboratory suggests that tasks with explicit references tend to be more sensitive to the effects of experimental manipulations such as transport delay.

A conservative guideline would state that a **total transport delay** of 200ms is acceptable in a low-level flight simulation. Recall, however, that any transport delay or equivalent delay in the simulation of the aircraft dynamics must subtracted from this delay budget. For a modern fighter aircraft such as an F-16, the **added simulator delay** should be limited to approximately 100ms for a low-level flight task.

Acknowledgments

References

1. Bailey, R. E. (1990, August). In-house Technical Memo on the F-16 C/D Simplified Transfer Function Model. (Buffalo,NY: Arvin Calspan Advanced Technology Center).

2. Diedrichs, R. W. (1990, January). Adverse Affect of Negative Gz on Subsequent High Positive Gz: A Need for Research and Education. Aeromedical and Training Digest, pp. 36-38.

3. Fischer, T. J., Riccio, G. E., & McMillan, G. R. (1986). The Effects of Simulator Delays on the Aquisition of Flight Control Skills. Proceedings of the 10th Symposium on Psychology in the Department of Defense . (pp. 224-228). USAF Academy: Department of Behavioral Sciences and Leadership.

4. Flying Qualities of Piloted Airplanes (1980, Nov). (1974). Military Specification (No. MIL-F-8785C). Wright Patterson AFB, OH: Aeronautical Systems Division.

5. Hall, R. (1989). Illumination and Color in Computer Generated Imagery. New York:

Springer-Verlag.

6. Johnson, W.V. & Middendorf, M.S. (1988). Simulator Transport Delay Measurement using Steady State Techniques. Proceedings of the AIAA Flight Simulation Technologies Conference. Washington, DC: American Institute of Aeronautics and Astronautics.

7. Kleiss, J. A., Hubbard, D. C., & Curry, D. G. (1989) Effect of Three-Dimensional Object Type and Density in Simulated Low-Level Flight (Report No. AFHRL-TR-88-66). Williams AFB, Arizona: Operations Training Division.

8. Kuo, B.C., (1987). Automatic Control Systems. Englewood Cliffs, NJ: Prentice-Hall, Inc..

9. Lusk, S.L., Martin, C.D., Whiteley, J.D., & Johnson, W.V. (1990). Time Delay Compensation Using Peripheral Visual Cues In an Aircraft Simulator. Proceedings of the AIAA Flight Simulation Technologies Conference. Washington, DC: American Institute of Aeronautics and Astronautics.

10. McRuer, D., Ashkenas, I., & Graham, D. Aircraft Dynamics and Automatic Control. Princeton, NJ: Princeton University Press.

11. Middendorf, M. S., Lusk, S. L., & Whitely, J. D. (1990). Power Spectral Analysis to Investigate the Effects of Simulator Time Delay on Flight Control Activity. Proceedings of the AIAA Flight Simulation Technologies Conference. (pp. 46-52). Washington, DC: American Institute of Aeronautics and Astronautics.

12. NASA Task Load Index (NASA-TLX): Paper and Pencil Package . (Version 1.0). Moffett Field, CA: NASA Ames Research Center

13. Phelps, M. (1984). Low Altitude Texture Comparison Data Base and Smooth Shaded Texture (Report No. AFHRL-TR-84-33). Williams AFB, Arizona: Operations Training Division.

14. Philips, C. A. & Petrofsky, J. S. (1983). The Mechanics of Skeletal and Cardiac Muscle. Springfield, IL: Thomas.

15. Riccio, G.E., Cress, J.D., & Johnson, W.V. (1987). The Effects of Simulator Time Delay on the Acquisition of Flight Control Skills: Control of Heading and Altitude. Proceedings of the 31st Meeting of the Human Factors Society. Santa Monica, CA:Human Factors Society

16. Vidulich, M. A., & Bortolussi, M. R. (1988). Speech Recognition in Advanced Rotorcraft: Using Speech Controls to Reduce Manual Control Overload. Proceedings of the American Helicopter Society National Specialists' Meeting - Automation Applications in Rotorcraft. Atlanta: Southeast Region AHS.

ISSUES CONCERNING CUE CORRELATION AND SYNCHRONIZATION OF NETWORKED SIMULATORS

Robert J. Sawler and Ron Matusof*, Staff Engineers

CAE-Link Corporation

Binghamton, NY

Abstract

To date, there has been only a limited effort to conduct research concerning cue correlation and synchronization of networked simulators. However, many of the basic concepts of simulator networking are similar to those faced by other industrial applications. These applications are diverse and include such areas as telecommunications, broadcasting, and robotics. The issues which must be addressed for each of these applications serve as a basis to understand the problems which may eventually face both the simulator network designer and user. Following a definition of cue correlation, this paper contrasts classical (single simulator) cue correlation with the requirements of simulator networks. Next it discusses research in related fields which have relevance to network cue correlation and identifies significant issues which must be addressed. Finally, avenues of research are proposed which should be pursued before acquisition of a production simulator network.

Introduction

The classical definition of dynamic response and cue synchronization has concentrated on the relationship between the control stick input and the kinesthetic cue responses of the simulated system. The close correlation of cockpit instruments, motion system, and visual system are representative of the fidelity of a simulator's dynamic response and cue synchronization performance. This performance is generally specified in terms of the latency between control inputs and the primary cues. The primary cues are determined along the axes of interest for a given simulator, and may be readily measured from the instrument, motion, and visual dynamic responses. By measuring these dynamic responses, all interactions which a crewmember can have with the simulated environment are evaluated. The correlation of the primary cues is extremely important to promote positive transfer of training and reduce simulator sickness.

Although the process is often tedious, difficult, and time-consuming, the techniques for measuring

dynamic response and cue synchronization performance for a single simulator have been well documented. The testing methodologies vary throughout the industry and include "pilot in the loop" stick inputs, high-energy inputs into the aerodynamic models, and even automated cue synchronization testing.

The recent interest in simulator networking has for the most part ignored the measurement of dynamic response and cue synchronization performance. The performance measurements of simulator networks involve many unique issues which have previously not been addressed. In fact, these issues are even unique among individual types of networks.

Interoperability between simulators can be grouped into three basic categories. First, a Local Area Network (LAN) can be operated within a single facility. Second, a LAN can be used to interconnect separate facilities of close proximity. Finally, a Wide Area Network (WAN) uses communications satellites, microwaves, and other transmission media to connect units at dissimilar locations. Each category of simulator network has a specific set of requirements and design limitations which impact the specification, implementation, and measurement of network dynamic response. The classical approach to cue correlation has been developed to ensure simulator dynamic responses, as specified by the appropriate regulatory or contracting agencies. There are no corresponding specifications which have been developed to optimize the dynamic cue responses inherent in distributed interoperable simulation.

The Networking Problem

The intent of networking two or more simulators is to provide a synthetic environment within which teams can interoperate. In general, the network is intended to provide enough fidelity so that the team operates in the synthetic environment as they would in the real world.

Fidelity is the degree of similarity, both physical and functional, between a training device and the actual equipment for which the training was undertaken.[1] Cue correlation provides a good indicator of fidelity since it is a quantitative measurement of the difference between the real world and the simulator.

---

* AIAA Member

Our understanding of what constitutes a well correlated system is based upon years of research into the problem of correlating the cues within a single simulator. Simulator networks introduce new concerns which require either the broadening of our cue correlation definition or adopting new standards for the networks.

Latency is a major concern for network cue correlation. Unlike the cues of a single simulator, the latency of individual cues on the network has a component which is both random and unbounded. Therefore, not only is the delay of an individual cue important, but the variation in the length of the delay is important as well. An additional latency concern involves closely coupled tasks, such as aerial refueling and remote weapon designation, which paradoxically require both closer correlation between cues and longer times for cues to be generated.

Latency of network cues is directly related to the speed of the network interface. The interface between the simulators must be of sufficient speed to allow the state of each simulator to be updated at a high enough rate so that all primary cues in participating simulators do not produce unrealistic or unstable results.[2] Large computational step sizes, due primarily to low data transfer rates, cause uncertainty in areas such as aircraft state and position. This is noticeable particularly at high maneuver rates. The resulting errors adversely affect crew responses, especially in high–gain, tightly coupled tasks.

Simulator cue correlation tests are primarily concerned with the dynamics of the simulation. For the simulator network, it is important to address the static correlation as well. For example, a cue which is normally not addressed in cue correlation tests is the positional accuracy of the simulator. When two or more simulators must traverse the same visual data base, positional accuracy of both the ownship and of objects within the data base becomes a significant correlation issue. Incorrect placement of objects in one player's data base can cause errors in execution of the simulated mission, potentially leading to a negative transfer of training.

Training systems are diverse in both their design and intent. Many envision that the networks of the future will connect dissimilar training systems which will interoperate within a common environment. Interoperable systems do not necessarily perform tasks identically or to the same level of fidelity, but the outputs of interoperable systems must be similar enough to provide the desired results. This means that the differences in computational units, computational rates, visual systems, interfaces, data bases, data base models, and the like must be accounted for. The correlation of cues between dissimilar simulators of differing fidelities is also an issue which must be addressed, since the fidelity of each participant simulator affects the overall correlation of shared cues.

Additional Primary Cues

The classical cue correlation approach recognizes the primary cues as the instrument, motion, and visual responses of the simulator. However, in the simulator network there are other primary cues which must also be addressed.

Navigation Accuracy

Contrary to the findings of Columbus (circa 1492), the earth is not spherical. Instead, the earth has a complex shape derived by combining ellipsoidal rotations around its spin axis. Simulator manufacturers generally implement an earth model which best fits the area of interest for a particular training objective. Therefore, different earth models are successfully used for dissimilar training objectives. When simulators are networked together, the dissimilarity of earth models produces navigational differences in the position of two objects, the distance between the two objects, and/or the direction of travel from one object to another. Team performance is adversely affected by these navigational differences since two team members flying the same course at the same speed may take different amounts of time to reach the same goal (i.e. positional errors) or they may end up in different places (i.e. directional errors) or both. The positional accuracy of each simulator, as well as the correlation of direction and distance measured at each network participant, is now a factor in the overall network cue correlation/synchronization scheme.

Audio Correlation

When an aural stimulus is perceived before a corresponding visual occurrence, even a short error of less than 20 ms can be detected.[3] Audio path delays within a single simulator have a negligible effect on the system's overall dynamic performance because these delays are always very small when compared with other system latencies. Most simulator network implementations use independent media for audio transmissions and network data transmissions. Thus, the audio may not be time coincident with the data it is describing. The delay actually occurs twice in normal conversation: once between the initiating simulator and the responding simulator, and once from the respondent back to the initiator. Recent studies indicate that the Minimum Audio Movement Angle

(MAMA) for audio detection from binaural systems may be a function of integration time.[4] The addition of unnatural delay may adversely affect some projected networked training tasks, such as the inclusion of dismounted infantry. The audio delay is bounded by a maximum delay which may never be exceeded. The maximum audio delay is a function of how many participants form a communications chain, where an increased number of participants, connected in series, causes an increase in the maximum possible delay.

"Wall-Clock" Time Accuracy

One of the most popular methods of ensuring data consistency across simulator networks is the use of time stamps. A time stamp is simply a data message containing the time at which data is placed on the network (or, alternately, the time at which the data was created). In theory, the receiving simulator can accurately calculate the network imposed delays by subtracting the time stamp from the current time. However, the synchronization of "wall clocks" is a significant correlation issue, since any synchronization signal sent across a large distance is itself subject to delays which are both significant and variable (although they are bounded). Any compensation for network delays which uses a comparison of non-synchronized clocks can introduce time-varying errors which accumulate as a mission proceeds. The accumulated errors tend to increase differences between the real world and the simulator, which corresponds to a decrease in system fidelity. A reduction in fidelity can reduce the amount of positive training transfer available on the network.

Network Latency

Latency is an issue of great concern to the simulator manufacturer. Most systems are designed to meet or exceed the Federal Aviation Regulation (FAR) Section 121 requirements established for airplane simulator qualification. The FAR requires instrument, motion, and visual system response to an abrupt control input of no more than 300 milliseconds after the airplane response (150 milliseconds is required for the higher fidelity categories of training).[5] Data which is passed between networked simulators must travel through long distances (when compared with the data within a single simulator) and through additional processing equipment. The draft military standard for distributed interactive simulation[6] contains the requirements for communication protocols between simulators. This standard recognizes the additional delays inherent in simulator networks and allows for

interconnection of simulators with up to 500 milliseconds of delay from end to end. Recent work at the Air Force Human Resources Laboratory[7] has shown that delays of this magnitude can be tolerated for air-to-air combat tasks provided that a first-order predictor is implemented (otherwise, 250 ms is the maximum tolerable delay). It is not currently required that the network cues meet the more stringent FAR Part 121 requirements. Within a single simulator, there are well documented delays which are typically accounted for in the cue correlation design of the simulator. These delays have been categorized into three mechanisms which account for their occurrence: data sampling, data transfer, and data processing.[8] These delays are shown graphically in Fig. 1. They are:

1. Control Loading Delay. This is the delay which a signal experiences between control input and generation of an output signal from the control loading system. In most simulators, this number is on the order of 20 ms.

2. Data Sampling Delay. This delay occurs due to the discrete nature of digital simulation. Inputs to a cuing system are sampled no faster than the effective iteration rate of the real-time host computer. A delay equivalent to one iteration of the host computer can develop when the control input occurs immediately after the host has sampled it (that is, the host computer will not know of the control input until the next sample). If the sampled data is passed between asynchronous systems, the data sampling delay may increase to include an additional delay caused by a "missed update" of the transfer.

3. Data Transfer Delay. This is the amount of time it takes to physically move the data from one memory location to another. Normally, this delay is insignificant since it is usually much smaller than the iteration rate of the host computer. However, when data is moved via a block memory transfer, such as a DR-11W interface, the data transfer delay can account for several iterations of delay.

4. Data Processing Delay. This delay represents the time it takes to operate on a sampled control input and produce a cuing output. Most modern simulators are designed to reduce this delay to a single iteration of the real-time host computer. The major cause of this delay is improper sequencing of the real-time software relative to data transfer times.

5. Visual System Delays. These are the delays associated with creating a visual image once a cuing output has been computed. This delay includes
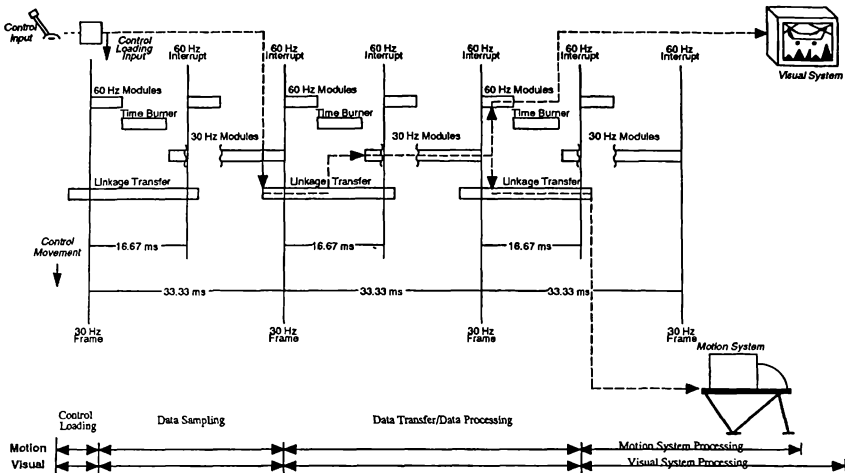
Fig. 1 Classical Throughput Delays for a Non-Networked Simulator (Typical)

both the data transfer delay and the data process-ing delay of the visual system.

6. Motion System Delays. These are the delays as-sociated with creating a kinesthetic cue once a cuing output has been computed. This delay in-cludes both the data transfer delay and the data processing delay of the motion (or other kines-thetic) system.

For a network of simulators, there are additional delays which must also be addressed. Each of these delays exists on an individual simulator, but are small enough that they can be considered to contribute a negligible effect toward classical cue correlation. These delays are explained in further detail as follows:

1. Network Transfer Delay. This delay is similar to the data transfer delay in that it represents the amount of time it takes to physically move data from a simulator to a network node. This delay can take up to several iterations of the simulator host computer. This delay also occurs at the re-ceiving node on the network, which may add up to several more iterations of delay.

2. Network Protocol Delay. This is the delay intro-duced in a data stream due to the choice of net-work protocol. This delay is not easy to measure, since it varies with the number of data packets passed, the number of network nodes, the trans-mission rate of the nodes, and the implementation of the protocol. Methods have been proposed for determining message and packet delays[9,10,11] but these appear to fall short for the heterogenous network structures which are envisioned for simu-lator networks.

3. Network Queuing Delay. This represents the delay which occurs as messages queue to be pro-cessed by a network node. The queuing delay occurs both when messages are transmitted to the network from a node and when messages are captured from the network by a node. The queu-ing delay appears to have a more significant effect when data is captured from the network. In fact, recent research indicates that message delay is invariant to the queue structure at the beginning of the path in many, though not all, cases.[12]

4. Network Transmission Delay. This is the delay in-troduced into a network due to its transmission medium. For LANs, this represents the charac-teristic delay of the network, which is typically around 20 ms for one complete traversal of the network.[13] For WANs, however, the transmission

430

delay is significantly longer. A satellite uplink and downlink introduces about 330 ms of delay under ideal conditions. Transmission via satellite is subject to propagation degradation from up to seven sources (galaxy, sun, ionosphere, atmosphere, hydrometers, terrestrial microwave, and earth).[14] Additionally, geosynchronous orbits of satellites are not perfect, and the satellite may move significantly (relative to allowable signal delay). The delay path through a satellite is therefore variable, based upon the weather, time of day, time of year, and other such factors.

The use of land lines does not necessarily preclude the variable nature of a long-distance data path. Terrestrial land lines and microwave facilities are generally owned by third parties who may switch the routing of the network data to another path without notice. This creates a data age uncertainty on the network equivalent to or greater than that experienced with satellites.

The choice of a transmission protocol often includes tradeoffs between system performance, number of network nodes, and cost. For example, Token Passing has been found to provide the best performance for military satellite communication networks, while Time Division Multiple Access (TDMA) has been found to provide the worst performance for the same application.[15] On the other hand, TDMA provides an efficient method of distributing data between large numbers of diversely located network nodes. The ultimate tradeoff for simulator network design will probably involve a compromise which sends data along multiple paths. Each path will then optimize the transmission of a particular type of data. While this is certainly a fine solution to network transmission problems, it does complicate the correlation process, since each transmission medium will introduce a unique latency to a subset of the network data.

5. Network Filtering Delays. This is the delay introduced by "smoothing" of asynchronous network state data updates. A major design consideration of all networks is the proper management of transmission bandwidth. One method of reducing bandwidth is to incorporate schemes involving prediction calculations, commonly referred to as "dead reckoning" algorithms. Prediction calculations extrapolate the state information of other networked entities between the network updates in order to reduce the number of times state information must be broadcast. Although they provide

a valid means of reducing network traffic, predictor algorithms also introduce data smoothing when network state updates occur. The smoothing is required to "dead-reckon" to the new network state without appearing to jump. The filtering delay represents the time to dead-reckon to a known state once the network has provided a new data sample.

Prediction calculations introduce a tradeoff between network traffic, the computational load of each participant simulator, and the precision with which each simulator perceives all other vehicles on the network. It has been shown that network traffic can be reduced by up to eighty percent by using a dead-reckoning algorithm.[16] However, this reduction in network traffic was accomplished by allowing vehicle appearance to vary up to three degrees in rotation and up to ten percent of the vehicle's dimensions in position before a state update is required. A measurable delay is introduced while the vehicle's appearance is dead-reckoned back to the known state. This delay becomes more pronounced as the amount of variation between the dead-reckoned and actual states increases.

6. Encryption Delays. This represents the delay introduced to a network of military simulators operating with classified information. Data must be encrypted prior to be transmitted by a network node and must be decrypted when captured by a receiving node.

7. Audio Delay. This is the total latency through the audio path. The audio delay is a measurement from audio input on one simulator to audio broadcast at the crew station of another. It is likely that this delay is negligible for all LANs, but may become significant across WANs, where it is possible that the audio will lead the simulator state. Studies have shown that the maximum tolerable end-to-end delay for audio is between 30 ms[17] and 50 ms.[18] The loss of audio synchronization with the state data of the aircraft could result in perceptible false cues in team interactions which could result in a poor transfer of training for tightly coupled high-gain tasks.

Delay Dispersion

Delay dispersion is the variation in the delay of a signal. It is generally not found on individual simulators because its prime causes are the asynchronous nature of the network, network unique delays, and the "floating" or variable data path that exists for WANs.

Delay dispersion must be accounted for when discussing network cue correlation. The variance in delay can cause a disordering of packets such that the sequencing of state information (player position and velocity, for example) is incorrect. To the crewmember, this will appear as a jumping or jittering effect in the primary cues. The resulting errors, if of significant magnitude, adversely affect crew responses, especially in high-gain, tightly coupled tasks.

Fidelity Differential

Networked systems have a degree of interoperability associated with them known as a "fidelity differential.[19] The fidelity differential determines the acceptable level of difference between the operations of two or more systems on the network and provides a method of expressing the required fidelity level relative to the required interactions of the systems. Cue correlation measurements for simulator networks will have to account for the fidelity differential and will have to show empirically that the primary cues are correlated within the allowable differential.

The designers of multifidelity networks must carefully implement their systems such that an unacceptable cue correlation error is not introduced to a higher-fidelity-level player because it is communicating with a lower-fidelity player. The fidelity differential can introduce correlation errors in three ways:

1. Fidelity of network data is insufficient to support the required cue correlation.

2. Update rate of network traffic is insufficient to support the required cue correlation.

3. Fidelity differential between representations of environmental models (navigation, terrain, etc) exceeds the tolerance of the required cue correlation.

Static Correlation

Static correlation refers to the correlation of individual simulator components in a non-real-time situation. This generally refers to the correlation of the terrain and cultural data bases within the visual system, radar system, map displays, navigation system, and threat portrayal of each individual simulator and between each simulator on the network.

Network cue correlation will require demonstration that position, distance, and direction between two objects located in the terrain or cultural data base are identical in each simulator on the network (within the

tolerance allowed by the fidelity differential). Additionally, the effects of visibility, time of day, and other environmental features could also be required for correlation testing, since each of these can adversely affect the team performance.

The Big Picture

The time delays introduced into a long-distance network are shown in Fig. 2. These delays may appear to be somewhat theoretical with no proven relevance to the training task. Fig. 3 shows the overall network simulator problem, and why cue correlation is especially important for tightly coupled tasks.

A control input is introduced into a simulator on a network. This control input is subject to the "classical cue latency" (Delay 1) which is commonly measured on a simulator: primarily instrument, motion, and visual dynamic response. Once the simulator's state has been updated, the new state is passed to another simulator on the network. The new state is delayed in transit by the delays which were previously discussed and arrives at the second simulator some time later. This delay is referred to as the network delay (Delay 2). The updated state of simulator 1 represents a control input to simulator 2. The control input is subject to the classical cue latency of simulator 2 (Delay 3). For coupled tasks (such as formation flight), the state of simulator 2 is updated in response to the control input from the network. The updated state of simulator 2 is transmitted to the network where it is subject to another delay (Delay 4). The updated state of simulator 2 now represents a control input to simulator 1, and it is subject to the classical cue latencies of simulator 1 (Delay 5). Thus, from control input to dynamic response on simulator 1 there are five delays, which may represent approximately 1.5 seconds of throughput delay, as shown below:

Delay 1 (Approximate) = 150 ms
Delay 2 (Approximate) = 500 ms
Delay 3 (Approximate) = 150 ms
Delay 4 (Approximate) = 500 ms
Delay 5 (Approximate) = 150 ms

These delays may exceed the latency tolerances of a task which a team intends to practice on the simulator network.

For a network with hundreds of players, many of which are driven by artificial intelligence rather than man-in-the-loop players, the problem becomes more complex. The application of artificial intelligence toward the semi-automated or fully-automated threat forces on simulator networks is quickly becoming real-
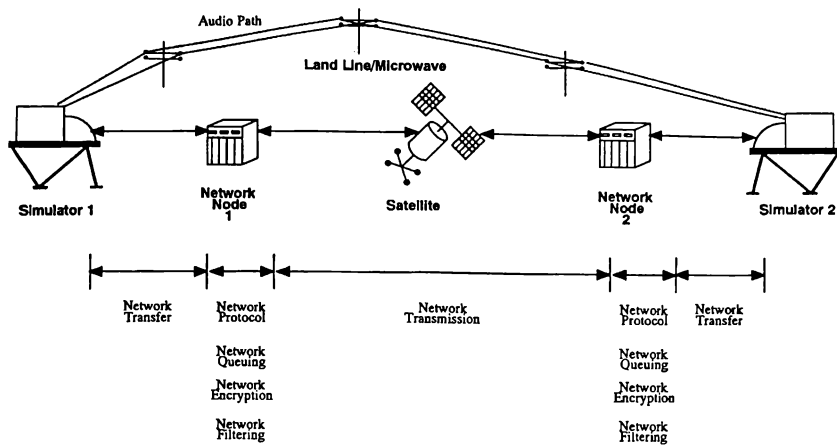
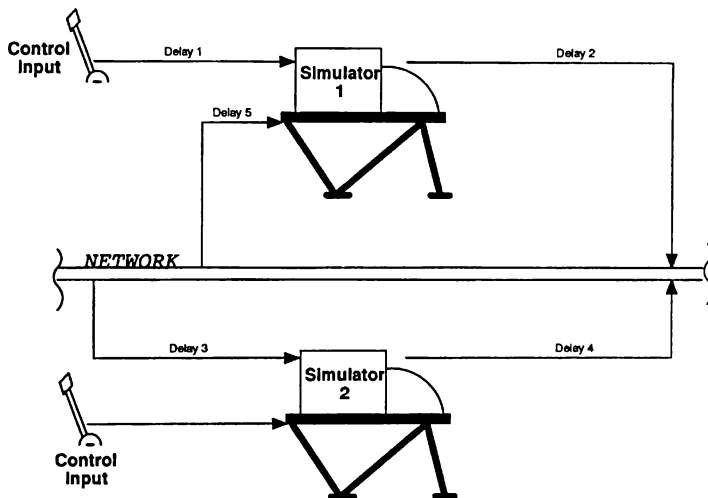Fig. 2 Time Delays for Long-Distance Networked Systems

Fig. 3 Simulator Networking Correlation Problem

ity. Artificial "thinkers" introduce another constraint to the network designer, since the "perceptions" of the artificial thinker are formed totally from data samples supplied by the network. The thinker stores perceptions for a number of frames before plotting a strategy, or next move. There is a wealth of related research which indicates that there is a 200 to 400 millisecond processing time required for artificially intelligent players to gather enough information in order to make a snap decision. For example, a robot playing Ping-Pong requires about eleven iterations of data samples in order to predict the ball's trajectory.[20] The robot requires additional time to replan its strategy and move its arms to intercept the ball with its paddle. Similar snap decisions are required by the artificially intelligent threat force. The latency of the network and the consistency of the state data are both important factors which will influence whether the threat makes a logical, tactically sound move or not.

Conclusions

Throughout the industry, cue correlation and synchronization have become recognized as one of the most important factors in providing quality synthetic flight training. Many years of research have been conducted to define and develop the requirements for individual training system cue correlation. The development of training system networking as a medium for training, evaluation, and rehearsal has opened many new avenues of research. This research has the potential of changing the way we think about correlation and synchronization.

Team operations involve a highly complex interaction between people and equipment, with each operating under very heavy workloads. The requirements for network cue correlation will be driven from both high-gain tasks and tightly coupled tasks. These requirements will introduce problems that did not even exist on individual training systems. At the same time, some of the traditional latencies may take on a reduced significance in the team training environment.

As of yet, there is not a standard which governs the cue correlation among networked simulators. Before any such standard can be developed, more research must be conducted to ascertain the effects that networking has on the cue correlation and synchronization.

Cue correlation and synchronization of non-networked simulators has a specified throughput delay which can be defined, calculated, and ultimately measured. The network issue introduces so many variables that each network will require rigorous upfront analysis and design to minimize the unnecessary delays as much as possible.

The overall performance of the network is only as good as its longest delay. Any efforts to produce high-fidelity cue correlation between networked simulators are wasted if the dynamics of the individual simulator are not addressed first.

References

1. Hays, R.T., "Simulator Fidelity: A Concept Paper", Army Technical Institute Technical Report 490, Alexandria, VA November 1980.

2. George, G.R., Knight, S.N., Stark, E.A. "Fidelity Requirements in Aviator Training Networks", AIAA Flight Simulation Technologies Conference, Boston, MA, September 1989.

3. Cooper, J.C. "Video-to-Audio Synchrony Monitoring and Correction", SMPTE Journal, September 1989.

4. Perrott, D.R., Tucker, J., "Minimum Audio Movement Angle as a Function of Signal Frequency and the Velocity of the Source", Journal of the Acoustics Society of America, April 1988.

5. Federal Aviation Administration, "Advisory Circular 120-40A: Airplane Simulator Qualification", US Department of Transportation, Washington, DC, July 1986.

6. "Military Standard (Draft): Protocol Data Units for Distributed Interactive Simulation", Institute for Simulation and Training, Orlando, Fl, January 1991.

7. Malone, H.L., Horowitz, S., Brunderman, J.A., Eulenbach, H., "The Impact of Network Delay on Two-Ship Air-To-Air Combat Simulation", AIAA Flight Simulation Technologies Conference, Monterey, CA, August 1987.

8. Cardullo, F.M., Brown, Y.J., "Visual System Lags: The Problem, The Cause, The Cure", IMAGE V Conference, Phoenix, AZ, June 1990.

9. Daigle, J.N., "Message Delays at Packet-Switching Nodes Serving Multiple Classes", IEEE Transactions on Communications, Volume 38, Number 4, April 1990.

10. Sklyarevich, I.K., "Analysis of Loss of Throughput in a Virtual Network Connection", _Avtomatika i Vychislitel'naya Teknika_, Volume 23, Number 2, 1989.

11. Golovko, N.I., "Message Delay Time at Network Node for the Case of Incoming Traffic of Variable Intensity", _Avtomatika i Vychislitel'naya Teknika_, Volume 23, Number 2, 1989.

12. Sushchenko, S.P., "Analysis of End-to-End Message Delay in a Multilink Virtual Circuit", _Avtomatika i Vychislitel'naya Teknika_, Volume 23, Number 2, 1989.

13. Seidensticker, S., "Multi-aircraft Simulation: Some Problems and Prospects", AIAA Flight Simulation Technologies Conference, Monterey, CA, August 1987.

14. Burkhart, R.M., "Propagation Phenomena and Terrestrial Interference in Satellite Television Transmission", 130th SMPTE Technical Conference, New York, NY, October 1988.

15. Koavennu, V.R., Aronson, M.H. Sites, M.J., "Protocols for Multisatellite Military Networks", IEEE, Monterey, CA, October 1986.

16. Miller, D., Pope, A., Waters, R., "Long-Haul Networking of simulators", 10th Interservice/Industry Training Systems Conference, Orlando, FL, December 1988.

17. Unemoto, K., Matsuo, N., "Voice Synchronization Methods for Burst Rate Coded Voice", IEEE Global Telecommunications Conference and Exhibition, Dallas, TX, November 1989.

18. Cooper, J.C., "Video-to-Audio Synchrony Monitoring and Correction", Society of Motion Picture and Television Engineers Journal, September 1988.

19. Knight, S.K., "Issues Affecting the Networking of Existing and Multifidelity Simulations:, Second Workshop on the Standards for Interoperability of Defense Simulations, Kissimmee, FL, January 1990.

20. Anderson, R.L., "Dynamic Sensing in a Ping Pong Playing Robot", _IEEE Transactions on Robotics and Automation_, Volume 5, Number 6, December 1989.

AN ASSESSMENT OF THE IMPORTANCE OF MOTION CUING BASED ON THE
RELATIONSHIP BETWEEN SIMULATED AIRCRAFT DYNAMICS AND
PILOT PERFORMANCE: A REVIEW OF THE LITERATURE

Frank M. Cardullo*
State University of New York
Binghamton, New York

Abstract

This paper reviews a method of assessing the need for motion cuing based on the simulated aircraft flight dynamics environment. The flight environment is reduced into four categories; maneuvers which are largely open loop and low gain, high gain closed loop with good visual, high gain closed loop with poor visual and aircraft which are unstable; and assesses motion cuing requirements on that basis. Also reviewed is the motion cuing literature including both the results of performance studies and transfer of training studies with the intent of establishing a determination of the relationship between the necessity of motion cuing and the task performed in the simulator.

Introduction

The topic of motion cuing in flight simulators is the most controversial issue in the community today. In fact, it has been so for quite some time, as was recounted by the inventor of the flight simulator, Ed Link, at the Royal Aeronautical Society 50 Years of Flight Simulation Meeting in London in 1979,

_____
*Associate Professor, Mechanical Engineering
Associate Fellow AIAA

related the following anecdote. Link stated that he was amused to find in reading a paper presented by the author at the Conference that motion simulation was still controversial. He was amused because he recalled having to make a rush trip to "Wright Field" in the forties to justify the necessity of motion on his trainers. Alas, I fear that it will ever be so.

There appear to be several reasons why it is controversial. The first is probably due to the fact that it does not enjoy face validity as do visual systems and some other simulator attributes. By this it is meant that it is obvious that a pilot can't make a visual approach to landing without the out-the-window scene displayed. However, it is not at all obvious that motion cues are necessary to accomplish that approach and landing. Another contributing factor to this controversy is the widely held view that, since the flight motion environment can't be replicated, there is no need to provide a reduced level of motion. It is interesting to note, however, that this argument is not applied to visual systems where a similar situation exists. For example, the human eye is capable of resolution on the order of one arc minute while most simulator

visual systems produce on the order of ten arc minutes except for area of interest displays which provide approximately three arc minutes resolution. Another example is the issue of brightness where differences of one to two orders of magnitude discrepancy exists with respect to daylight conditions.

Another indictment of motion systems is that they are expensive to purchase, to house, and to operate. In a study conducted by the RAND Corporation for the Air Force C-17 training program (Gebman et al. 1986), it was determined that the motion system would add four percent to the 25 year life cycle costs of the device. That figure included facility costs and was based on a conventional, state-of-art hexapod motion system.

Visual system constraints are often cited as a reason for not employing platform motion systems. This concern is not without a technical basis. For perceptual reasons it is desirable to have visual images appear to be located at infinity. This can be accomplished by using collimating optics or a real image display system with a screen such as a dome. It is desirable, if a real image system is used, to present the image as far away as possible. Placing this image 10 feet from the eye is generally considered the absolute minimum eye relief distance while 20 feet is felt to be optimal. If platform motion is to be used, this dome is generally mounted on the platform. The practical upper

limit for a dome which can be attached to a conventional hexapod motion system is twelve feet in radius. Therefore, the use of motion then constrains the visual display options. There are other visual solutions such as collimated displays or using a large fixed dome with the cab on a motion system inside the dome. This later solution may require that the visual image generator have the capability of dynamic non-linear image mapping.

The opponents of employing motion systems on training simulators cite that no basis in transfer of training studies can be found. To some extent this is true, however, most of these studies were conducted with inferior equipment. This issue will be elaborated upon later.

This paper first presents a discussion of the motion environment encountered in flight, characterized with respect to maneuver categories and aircraft handling qualities. The next section reviews the relevant research which is primarily pilot performance based. The last section summarizes and draws conclusions. Those who desire an in-depth treatment of motion perception are referred to Borah et al. (1977), Hosman and van der Vaart (1980), and Martin (1991).

Categories of Aircraft Motion

It seems reasonable to assume that motion requirements may be dependent on the extant motion environment. The de-

gree to which this relation-
ship holds is perhaps debat-
able, however, it does provide
a rational context within
which an analysis may be per-
formed. Some authors (Martin
1990) have concluded that
motion cuing may only be nec-
essary in disturbance motion
situations, which is a depar-
ture from the standard USAF
view that motion is not re-
quired in any centerline
thrust aircraft. It should be
noted here that the above two
views are stated in the train-
ing context.

It is useful at this point to
provide a taxonomy of the
aircraft motion environment.
Aircraft motion has been di-
vided, by Hall (1989) and
others, into two categories,
disturbance motion and maneu-
ver motion. These two catego-
ries can be further subdivid-
ed. First consider distur-
bance motion. Disturbance
motion is characterized by its
stochastic nature, and is due
to external, usually atmo-
spheric, conditions. It may
be either random continuous or
random discrete. The random
continuous disturbances are
things like buffet, and "rough
air". Whereas the random
discrete disturbances include
phenomena such as wind shear,
wind gusts, and atmospheric
pressure discontinuities.

Maneuver motion can be further
separated into pilot initiated
discrete changes in flight
path, pilot continuous control
input required for precision
tracking in high gain closed
loop control tasks and pilot
continuous control of a vehi-
cle with low stability. The

first category is exemplified
by maneuvers such as heading
changes and altitude changes.
The second may be illustrated
by formation flying, ground
attack, etc. The final cate-
gory of control of low stabil-
ity aircraft is experienced in
helicopter piloting and air-
craft such as forward swept
wing, tilt rotor and vectored
thrust vehicles (Harrier).
Wind shear, which was previ-
ously described as discrete
disturbance motion, is in
reality a combination of dis-
crete disturbance motion when
the shear is encountered, and
maneuver motion in controlling
the aircraft through the dis-
turbance.

Simulation Purpose

It is not only the aircraft
motion environment which dic-
tates the requirement for non-
visual motion cuing, but also
the purposes for which the
simulator will be used must be
considered. Some examples of
simulator utilization are;
workload measurements, perfor-
mance analyses, handling qual-
ities assessment, cockpit
display development and air
traffic control (ATC) integra-
tion. These are all engineer-
ing and R&D purposes but
flight simulators are also
used extensively for training.
It should be pointed out at
this point that the metric for
evaluation applied to training
simulation has been transfer
of training, whereas the
metric applied to R&D simula-
tors has been performance.
There are some who believe
that performance metrics are
appropriate in both cases,
particularly since it is ex-

tremely difficult to conduct valid transfer of training studies.

It is of interest to note at this point that most disturbance motion can be provided by small amplitude, in-cockpit devices such as seat vibration systems. An exception to this would be wind shear where a more comprehensive assessment of the motion environment is necessary in order to apply the correct recovery strategy. In some cases, the onset of a phenomenon can be considered to be in the disturbance category, but the recovery from it would be maneuver motion and usually of the high gain, closed loop type. Examples of this would include failure modes and the previously mentioned wind shear encounter.

The previously enumerated motion categories were related to simulation motion cuing requirements in a Royal Aeronautical Establishment technical memorandum (Hall 1989). In this work, the author contends that non-visual motion cues are of little importance in largely open-loop, low gain, low workload maneuvers with strong visual cues. It should be noted that these would also tend to be lower frequency maneuvers which are in the more sensitive portion of the visual detection of motion and less sensitive range of the vestibular system (Brown, Cardullo, Sinacori (1989). According to Hall (1989), motion cues are more important for high workload, higher gain maneuvers, especially with poorer visual cues, necessary for high gain

tasks even with good visual cues and essential for control of low stability vehicles.

It has been reported that the onset of vection is substantially delayed in the absence of vestibular or somatosensory cuing. Several researchers (Berthoz et al, 1975, Dichgans 1980+, Young et al 1973) have found that, in the absence of vestibular stimulation, the onset of vection requires from 10 to 30 seconds. However, with only minimal vestibular stimulation such as a jolt in the correct direction, vection onset is reduced to tenths of seconds. The vection latency issue seems to provide a compelling argument for at least some vestibular stimulation in a flight simulator under some conditions. Therefore, it seems that non-visual motion cuing is necessary to alleviate this delay which the next section will illustrate is detrimental to proper control of the simulated air vehicle.

Motion Performance Studies

This section presents the results of several studies in which the contribution of non-visual motion cuing can be assessed. The review is not comprehensive, but represents a sampling sufficient to illustrate the points.

An experiment conducted at NASA Ames on the FSAA (Showalter & Parris 1980) illustrates the effect of motion on two piloting tasks. The first task required the pilot (n=48) to respond quickly with appropriate rudder and wheel
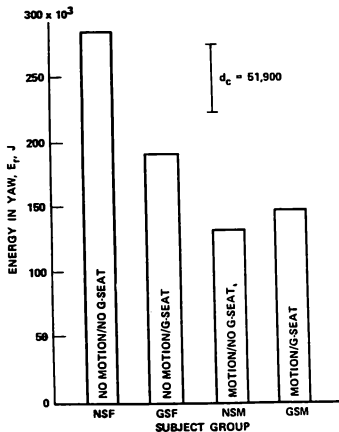
Figure 1

inputs to recover from an engine out on takeoff. Figure 1 illustrates the differences in performance for the four cuing conditions. All four conditions include visual cues and the four motion conditions were; fixed base, g-seat only, platform motion only and platform motion with g-seat. It can be seen that all active motion conditions offered statistically significant reductions in yaw energy over the fixed base case.

The performance metric employed was yaw energy which indicated how quickly the pilots were able to recover the airplane and with how little control activity. Therefore, it is obvious that their performance deteriorated without motion cuing.

Figure 2 illustrates the ef-

fects of motion on a pilot's ability to make precision turns. These data also show the effect of pilot experience, the "high time" pilots averaged 1484 hours flight time and the "low time" group averaged 270 hours. Here it is seen that there is no significant effect of motion on the low time pilots, however, there does appear to be significant effect on high time pilots. While normally turning flight might be considered low gain, in this case the pilots were required to maintain precise control of bank angle, airspeed, altitude and heading. These data illustrate the benefit of platform motion cuing in particular, but less benefit due to g-seat implementation. One possible explanation for this is that the g-seat employed here was designed to provide sustained acceleration cues in a fighter aircraft while the aircraft simulated in this experiment
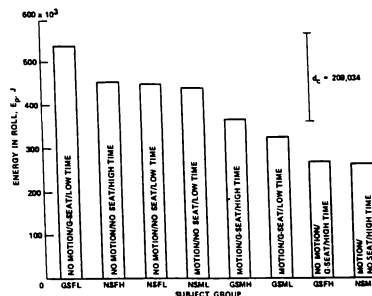

Figure 2

was a KC-135. It should also
be noted that the g-seat used
in these experiments had a
bandwidth of less than one
hertz.

A similar study was conducted
by Parris and Cook (1978) at
NASA Ames employing the same
simulator and the same simu-
lated aircraft. The maneuvers
employed were engine failure
before liftoff and engine
failure after liftoff. They
examined four cuing condi-
tions, all of which included
cockpit instruments and sound
systems. The first condition
included only instruments and
sound, the second added visu-
al, the third removed visual
and added motion, and the
fourth used all four cuing
systems.

The results indicated that the
first condition was the worst,
adding visual resulted in a
significant improvement, re-
moving visual and adding mo-
tion was significantly better
than instruments alone but not
as good as visual, especially
in the case of the failure
before takeoff. Visual cuing
is essential to keep the simu-
lated aircraft on the runway.
The best performance condition
was with motion and visual
cuing.

In two experiments reported on
by Young (1967) considerable
improvement in pilot perfor-
mance was noted with the addi-
tion of motion cues. The
first experiment involved a
pilot attempting to recover
from a lateral disturbance
while on the glide path.
Figure 3 illustrates a
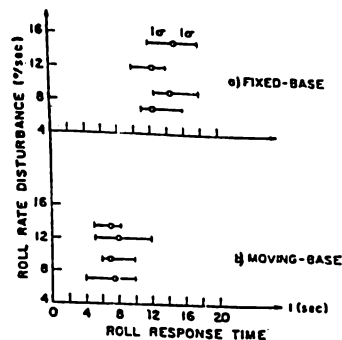statistically significant

Figure 3 Median roll-response to
step roll-rate disturbance in simu-
lated landing.

reduction in the recovery
interval when motion cues are
available. The data presented
include four levels of distur-
bance.

The second experiment required
pilots to hover an uncompen-
sated helicopter. Three pilot
groups were used in this ex-
periment: helicopter pilots,
highly trained non-pilots, and
moderately trained non-pilots.
It can be seen in Figure 4
that for the helicopter pilots
and the highly trained non-
pilots, their performance
improved substantially with
the addition of motion cues to
the visual.

Another study which demon-
strates the effects of motion
cuing on simulator pilot per-
formance was performed at
Delft University of Technology
in the Netherlands (Hosman &
van der Vaart 1980). In this
study, the interaction of

vestibular and visual stimuli and the influence of foveal and peripheral visual as well as vestibular cues on the control behavior of a subject in a roll axis task were examined. Two tasks were used in the experiment: the first was a disturbance regulation task, and the second was a tracking task. Both tasks were roll axis specific and the dynamics were that of a double integration. With this model of the controlled system, there is no opportunity for coordinated maneuvers, hence, there will be a side force component generated by the motion system due to tilt. Since the motion system used only has three degrees of freedom, there is no way to compensate the tilt cue.
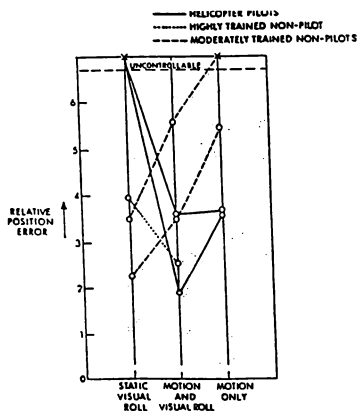

Figure 4 Effect of motion and visual roll cues on helicopter lateral hovering.

Figure 5 illustrates the standard deviation of roll angle for both the disturbance and following tasks. Notice that for the disturbance task, when peripheral displays are added to the central, pilot performance improves by about 10%, then when motion cuing is added to the foveal visual, the performance improvement is about 60%. The further addition of peripheral visual improves the situation only marginally.

Motion alone and peripheral vision with motion are slightly worse. The worst case is peripheral displays without central vision. Similar trends are seen in the following task.
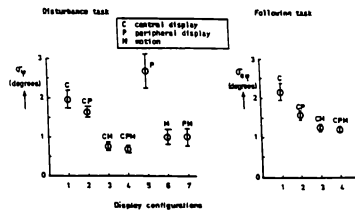

Figure 5 Performance of disturbance task and following task as a function if display configuration.

Figure 6 presents the crossover frequency and the phase margin for each of the conditions and for each of the two tasks. These terms perhaps require some explanation in the context of this paper. The crossover model as developed by McRuer and Krendel (1974) is used frequently to describe pilot performance. It is well known that, quite independent of the actual aircraft dynamics, the pilot

adapts his control behavior (system gain) such that the gain crossover frequency, is between three and five radians per second, and his phase margin, is between 20 and 40 degrees. The pilot can trade off one for the other, i.e. phase margin for gain or vice versa. If the phase margin gets too low, the system becomes unstable. This notion is supported by a very large body of literature.

Examination of Figure 6 indicates that the addition of vestibular motion cues increases the crossover frequency to about five radians per second while the phase margin stays at about 15 to 20 degrees, for the disturbance task. The tracking task presents some ambiguous results, however, namely, that the phase margin keeps increasing to over 50 degrees with the addition of motion cues while the crossover frequency decreases. This result is curious in light of the performance improvement shown in Figure 5.

One of the most interesting studies relating the value of motion cuing to performance was one undertaken by the Armstrong Aerospace Medical Research Laboratory (McMillan et al 1985). The study compared the performance of pilots in a simulator employing a dynamic seat as a motion cuing device and a full motion device, the roll axis tracking simulator (RATS). The task was a single axis (roll) disturbance regulation activity. A narrow field of view visual display presenting a graphic illustration of

aircraft roll attitude was used. The dynamic seat is a high bandwidth (10Hz) device, designed to conduct research
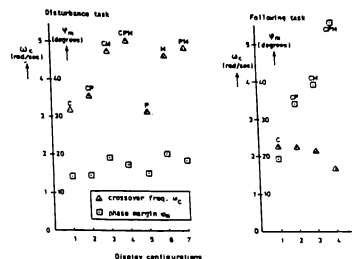

Figure 6   Phase margins and crossover frequencies of $H_p(s).H_c(s)$ as a function of display configuration.

in providing motion cues with an in-cockpit device. Figure 7 summarizes the results of this study which combined both performance metrics and the transfer of training criterion.

The graph presents a plot of mean RMS error vs training session. The error is the difference between the aircraft bank angle and wings level for any run. The pilots performed this task until their performance reached asymptote. The curve labelled full motion is the task performed in the RATS while the "static" case is the pilot flying the simulator with the dynamic seat inactive. The three seat active curves represent three different seat drive algorithms. The "seat-velocity" algorithm employs a seat position proportional to

simulator aircraft roll rate paradigm, "seat-position" makes dynamic seat angle proportional to simulated bank angle and 'seat-sigma" causes
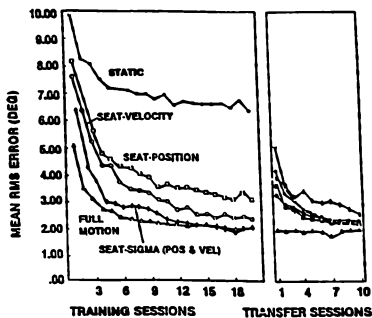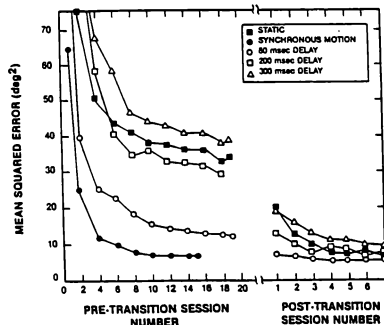

Figure 7

the seat position to be proportional to a linear combination of aircraft roll rate and bank angle.

It is clear that all three algorithms offer considerable improvement over the static case with the "seat-sigma" approach being quite close to the full motion criterion. While this result is quite encouraging from the performance metric perspective, it is somewhat disappointing in its training results. It can be seen in the transfer of training portion of the graph that there is only a slight improvement in the dynamic seat cases over the static case.

An interesting result was achieved in an experiment on the effects of time delay on

motion cuing in which the same dynamic seat and same task were used (Levison, Lancraft and Junker 1979). Figure 8 presents the data for this study which illustrates that the case where motion cues are delayed 300 ms with respect to the visual produces poorer performance than the static case. The 200 ms delay case offers only marginal improvement over the no motion condition while the 80 ms asynchrony shows substantial improvement. Hence, these results illustrate at least one case where "bad" motion is worse than no motion.

Previously, it was mentioned that where motion seems to have the least effect is in training. This was somewhat borne out in the data presented in Figures 7 and 8. There were two previous studies (Waters et al 1976) and (Grey and Fuller 1977) which showed no training benefit from motion cuing in the tasks inves-

EFFECTS OF DELAYED MOTION CUES ON LEARNING, PERFORMANCE, AND TRANSFER-OF-TRAINING. (From Levison, Lancraft, and Junker. 1979)

Figure 8

tigated These latter two stud-
ies have subsequently been
largely discredited because
they were performed using
inferior motion hardware with
long cue delays, a suspect
cuing algorithm and a software
update rate of 7.5 Hz as com-
pared to 30 to 60 Hz which
current simulators employ.

Conclusions

As was stated at the outset,
there exists a great deal of
controversy over the necessity
of motion cuing in flight
simulators and it is not ex-
pected that this brief summary
will neutralize the entire
controversy. However, it is
hoped that the major issues
have been illuminated from a
technical perspective. Howev-
er, it can be stated that most
negative decisions about using
motion systems are based not
on motion requirements analy-
ses but rather on anecdotal
references, intuition, inaccu-
rate cost assessments or the
imposition of visual limita-
tions. With respect to the
cost issue, the RAND Study
previously cited, indicated
that motion systems, including
facilities, added four percent
to the life cycle cost of a C-
17 training simulator. Hence,
cost does not appear to be a
major issue. The constraints
on visual system design, as
previously discussed, comes
closest to a real technical
issue. However, resolution of
this problem is fairly easy in
most cases.

Where some transfer of train-
ing studies were done to eval-
uate the effect of motion on
training, most were done with

inferior equipment, low com-
puter frame rates and/or poor
cuing algorithms. An excep-
tion to this may be the work
done at the Armstrong Aero-
space Medical Research Labora-
tories. However, these exper-
iments did not use a motion
platform but a high bandwidth
dynamic seat. The seat may
not have sustained the cue for
a sufficient time to have a
training effect and the visual
scene was rudimentary with a
very narrow field of view.
The experiments are currently
being repeated with a wide
field of view visual system
with much more scene content.
Probably the only conclusions
which can be drawn from most
of the training research is
that bad motion may be worse
than no motion.

The body of literature using
pilot performance as the cri-
terion, which is the appropri-
ate metric for engineering
simulation, shows a positive
effect for tasks where it is
expected that there would be
an effect and even some where
no motion effect was expected
based on the flying task or
the purpose of the simulation.
The flying task to motion
cuing requirements relation-
ship runs the gamut from es-
sential for low stability
vehicles; to necessary for
high gain tasks even with good
visual cues; to important for
high gain high workload maneu-
vers, especially with poor
visual cues; and of little
importance in low gain, large-
ly open loop tasks.

Finally, the argument was made
that to ensure minimal laten-
cies in the onset of the visu-

al sensation of motion that some vestibular or somatosensory motion cuing is necessary. This may be particularly true to minimize the occurance of simulator sickness. The dynamics of the aircraft and task are important from this perspective as well. For low frequency low workload situations the perception of motion is dominated by vision, therefore, vestibular and/or somatosensory cuing is not necessary for the illusion of motion.

Hence, the bottom lines are: Is motion necessary? Sometimes! It depends on flying task, simulation purpose, and psychophysiological requirements. It is probably necessary more often than the conventional wisdom dictates. If motion is provided, it should be "good" motion because "bad" motion is worse than no motion.

More training transfer studies should be undertaken to provide direct quantitative evidence of the need for non-visual motion cuing. In addition, the development of a basis for extending the performance results to training simulators is also needed.

REFERENCES

Berthoz, A., Pavard, B., Young, L.R. "Perception of linear horizontal self-motion induced by peripheral vision (linear vection); basic characteristics and visual-vestibular interactions." Exp. Brain Res. (1975) 23:471-489.

Borah, J., Young, L.R., Curry,

R.E. "Sensory mechanism modeling." AFHRL-TR-77-70. October 1977.

Brown, Y.J., Cardullo, F.M., Sinacori, J.B. "Need based evaluation of simulator force and motion cuing devices." Paper 89-3272. AIAA Flight Simulation Technologies Conference, August 1989, pp. 78-85.

Dichgans, J. "Visual-vestibular interaction in roll: psychophysics and physiology." Source unknown. Author's address, Department of Neurology, Liebermeisterstr, 18-20 Universitat, 7400. 1980+.

Gebman, J.R. et al. "Assessing the benefits and costs of motion for C-17 flight simulators: Technical Appendixes." R-3276-AF RAND Corporation. June 1986.

Grey, T.H., Fuller, R.R. "Effects of simulator training and platform motion on air-to-surface weapons delivery training." AFHRL-TR-77-29. July 1977.

Hall, J.R. "The need for platform motion in modern piloted flight training simulators." Technical Memorandum FM35, Royal Arospace Establishment. October 1989.

Hosman, R.J.A.W., van der Vaart, J.C. "Thresholds of motion perception and parameters of vestibular models obtained from tests in a motion simulator. Effects of vestibular and visual motion perception on task performance." Memorandum M-372, Delft University of Technolo-

gy, Department of Aerospace Engineering, Delft, The Netherlands. August 1980.

Levison, W.H., Lancraft, R.E., Junker, A.M. "Effects of simulator delays on performance and learning in a roll axis tracking task". AFDL-TR-79-3134. WPAFB, Ohio. 1979.

Martin, E.A., Osgood, R.K., McMillan, G.R. "The dynamic seat as an onset cuing device." Proceedings AIAA Flight Simulation Technologies Conference, August 1987, Monterey, California.

Martin, E.A. "Motion Cuing I, Lecture Notes." Flight Simulation Update, State University of New York at Binghamton, Binghamton, NY. January 1990.

McMillan, G.R., Martin, E.A., Flach, J.M., Riccio, G.E. "Advanced dynamic seat: an alternative to platform motion." Proceedings 7th Interservice/Industry Training Equipment Conference Nov 19-21, 1985, pp. 153-163.

McRuer, D.T., Krendel, E.S. "Mathematical models of human pilot behavior". AGARDograph No. 188, 1974.

Parris, B.L., Cook, A.M. "Effects of visual and motion simulation cuing systems on pilot performance during takeoffs with engine failure." NASA TP 1365. December 1978.

Showalter, T.W., Parris, B.L. "The effects of Motion and g-seat cues on pilot simulator performance of three piloting tasks." NASA TP 1601. January 1980.

Waters, B.K. et al. "Preliminary investigation of motion visual and g-seat effects in the advanced simulator for undergraduate pilot training (ASUPT)." Proceedings 1976 AIAA Visual and Motion Simulation Conference.

Young, L.R. "Some effects of motion cues on manual tracking." Journal of Spacecraft Vol. 4, No. 10:1300-1303. October 1967.

Young, L.R., Dichgans, J., Murphy, R., Brandt, Th. "Interaction of optokinetic and vestibular stimuli in motion perception." Acta Otolaryng 76:24-31. 1973.

THE NASA LEWIS INTEGRATED PROPULSION AND FLIGHT CONTROL SIMULATOR

Michelle M. Bright[*]
NASA Lewis Research Center
Advanced Controls Branch
Cleveland, OH 44135

Donald L. Simon[**]
US Army Aviation Systems Command
Propulsion Directorate
Lewis Research Center
Cleveland, OH 44135

ABSTRACT

A new flight simulation facility has been developed at the NASA Lewis Research Center in Cleveland, Ohio. The purpose of this flight simulator is to allow integrated propulsion control and flight control algorithm development and evaluation in real time. As a preliminary check of the simulator facility capabilities and the correct integration of its components, the control design and physics models for a Short Take-Off and Vertical Landing fighter aircraft model have been demonstrated, with their associated system integration and architecture, pilot vehicle interfaces, and display symbology. The initial testing and evaluation results show that this fixed based flight simulator can provide real-time feedback and display of both airframe and propulsion variables for validation of integrated flight and propulsion control systems. Additionally, through the use of this flight simulator, various control design methodologies and cockpit mechanizations can be tested and evaluated in a real time environment.

INTRODUCTION

Historically, the NASA Lewis Research Center has been involved with the design, evaluation, and testing of control designs for advanced engine concepts. Future advanced aircraft configurations, however, will require the integration of the propulsion control system with the flight control system. The Advanced Controls Technology Branch at NASA Lewis is conducting research in this area of integrated flight and propulsion control design, specifically for a Short Take-Off Vertical Landing (STOVL) aircraft. This integrated control design effort requires a means to test and evaluate these integrated control design algorithms. The flight simulator facility developed in this study provides a means to validate integrated design methodologies, to monitor, in real time, engine and airframe parameters during simulation, to evaluate new software partitioning methods, and to test control specification bandwidths and control rates during piloted engineering evaluations.

This integrated propulsion and flight control simulator is an evaluation station for flight and propulsion control research consisting of a cockpit, displays, and visual out-the-window scenery. This paper describes this flight simulation environment; the system communications to integrate the visual system with the host simulation computer and the control computer; the control design environment; the development for an integrated control task flight simulator cockpit effectors and displays; and simulation testing of the flight simulator using a STOVL aircraft model and integrated control design. Finally, conclusions concerning the suitability of the flight simulator to current research, and recommendations for future enhancements are given.

SIMULATION ENVIRONMENT

The flight simulator facility, as shown in Figure 1, consists of five major components. The Paragon Graphics Visual System generates the Heads Up Display (HUD), the Heads Down Display (HDD), and the out-the-window scenery. The single channel projection system displays the scenery information and the HUD symbology. A mockup fighter cockpit provides pilot effectors for the control of engine and airframe commands. The Applied Dynamics System 100 real time simulation computer executes the real time engine and airframe simulations. Finally, the control computer

---

* Electrical Engineer, Member AIAA.
** Electrical Engineer.

system executes the integrated control design algorithms.

Visual System

The visual system consists of an image generation processor and a PC 80386-20 development station. The image generation processor provides 40 degree by 50 degree color database images with a screen refresh rate of 60 hertz and a scene content of 2000 polygons. Image resolution is 1024 by 1024 pixels. The resolution of the image processor is 32 Bit floating point, and the database can provide up to 16 moving models at any one time. [1]

The development station allows for development of new software models. Additionally, the development station serves as a run time front-end to the visual system. The development station is a 20 MHz, 386-based PC with a 40 Megabyte hard disk, a 5 1/4 inch floppy disk drive, monitor, keyboard, and a chassis with 8 expansion slots. The development station has a DR11W digital parallel interface adapter for communication with the both the control computer and the simulation computer. All the image database management software and image generation libraries for both HUD and HDD development reside on the development station. The development station supports C, FORTRAN, and assembly language programs.

Projection System

The forward projection system consists of a single channel, color projector with a 40 degree in the vertical by 50 degree in the horizontal field of view. The out-the-window scenery and heads up display information are displayed on a free standing, curved, high resolution projector screen.

Cockpit

The mockup fighter cockpit contains a four position, spring loaded, sidestick controller, linear motion sliding throttle, rudder pedals, and a color touch screen monitor to emulate heads down instrumentation. Several discrete switches on the sidestick controller and linear throttle can be used to simulate speedbrake, mode switching, or weapon related functionality.

Control Computer System

The control computer system consists of Control Interface and Monitoring Unit (CIM) with a control microcomputer, interface hardware, and a hardware and software monitoring system. The CIM unit was fabricated in house to implement and evaluate advanced digital control algorithms with hardware in the loop. It consists of a microcomputer with a real time operating system, the interface hardware for connecting to an engine/airframe simulation or actual engine, and monitoring hardware and software to verify that the control is performing properly. The control computer is intended for use in both simulation and engine test facilities, and is therefore implemented in a portable equipment rack. [2]

The integrated flight and propulsion control algorithms are executed in the control microcomputer of the CIM unit. It consists of a 20 MHz 80386 single board computer, analog and discrete I/O boards, and disk drives with their associated controller boards. The circuit boards are mounted in an industry standard Multibus I microcomputer chassis. The microcomputer runs the iRMX real time operating system. This operating system provides the services needed to construct a real time executive to schedule the execution of the integrated control algorithms, I/O routines, and the data collection software. The executive is coded in PL/M and uses timer generated interrupts to update the control at the desired interval. The integrated control algorithms executing on the microcomputer are coded in FORTRAN.

The purpose of the interface hardware is to route signals throughout the control computer, to connect the control computer to external devices, and to buffer those signals if desired. Cables which interface to both the simulation computer and the cockpit are terminated at the control computer base connectors. A patch panel is available to control the routing of signals throughout the system and to allow quick changes in configuration. The control computer is capable of supporting 64 analog inputs, 32 analog outputs, 24 discrete input signals, and 32 discrete output signals.

The monitoring capabilities of the control computer allow the user to observe analog signals that are sent to and from the system, as well as record variables within the control algorithms during execution. A data acquisition system monitors the control computer I/O and allows the operator to display any desired signal or signals in actual voltages or in engineering units. The Microcontroller Interactive Data System (MINDS) program runs in the spare time on the microcomputer CPU. The MINDS program has both steady-state and transient data gathering capabilities and can access any variable in the control algorithm for display or plotting. These monitoring capabilities allow the user to insure proper control operation and acquire data for later analysis.

Simulation Computer System

The airframe and propulsion models are implemented in the simulation computer which is specifically designed for real time and time critical simulation of continuous, dynamic systems. The simulation system consists of an Applied Dynamics System 100 simulation computer, analog and digital I/O hardware, and a VAXstation II front-end computer. The simulation computer is a 64-bit floating-point multiprocessor which is optimized for numerical integration. The I/O facilities allow communication with the integrated control algorithms running on the control computer, and allow the updated engine and airframe parameters to be transferred to the visual generation system. The models implemented on the simulation computer are coded in ADSIM, a proprietary programming language. Data collection and graphical display utilities are available to allow the user to monitor the simulation. The simulation computer at NASA Lewis is capable of supporting 44 analog input signals, 44 analog output signals, and 32 discrete input and output signals. [3]

SYSTEM COMMUNICATIONS

Design of the flight simulator system configuration, interfaces, and mechanization of the cockpit and displays was performed at NASA Lewis. A system diagram with interface interactions is given in Figure 2. This figure shows that the cockpit control effectors, (i.e. sidestick, throttle, rudder pedals, thumbwheel), produce pilot commands in the form of analog signals. These analog signals are sent directly to the control computer where the control algorithms process the commands. Any discrete commands from the cockpit (switches) are passed over discrete lines directly to the control algorithms on the control computer. Analog control commands generated by the control algorithms on the control computer are sent to the engine and airframe simulations which reside on the simulation computer. Engine and airframe data are passed back to the control algorithms on the control computer via analog signals. Updates in airframe and engine parameters are passed through a digital parallel DR11W interface to update the visual system, heads up display, and heads down display.

The DR11W interface consists of a circuit board in each of the computers connected by two 40 conductor flat ribbon cables 50 feet long. Programming of the DR11W consists of manipulating available registers to implement the handshaking needed to transfer data between the computer systems. A C language program on the visual system side, and an ADRIO language program on the simulation computer side are used to control the DR11W interface.[4] Since data are stored on the computers in different 32 bit floating point formats, the data on the simulation computer has to be converted to the visual system format prior to data transfer. After the conversion, the 32 bit floating point data is split into two 16 bit words which are transferred over the interface individually and recombined in the visual system. It requires approximately 1 millisecond to transfer 7 floating point numbers over the DR11W interface.

CONTROL DESIGN ENVIRONMENT

The integrated flight and propulsion control algorithms which are evaluated on the flight simulator are developed in an automated control design environment called MATRIXx, which runs on a VAXstation computer. Within this environment the designer can graphically assemble a block diagram representation of the control system and analyze the representation for

correct operation.

Various analysis, design, and optimisation tools are available to permit control design and verification. After the control design is complete, a code generation utility is used to generate a FORTRAN version of the discrete time controller from the block diagram representation. This code generation utility is also capable of producing source code in ADA and C programming languages. [5]

The FORTRAN code that is generated from this process is downloaded to the control computer. After compilation the control design is executed for evaluation within the flight simulation facility. If changes to the control are deemed necessary, the above procedure can be repeated.

COCKPIT EFFECTORS AND DISPLAYS

Development of the Pilot Vehicle Interfaces (PVI) for this flight simulator was based upon PVI research by Merrick, Farris, and Vanags at NASA Ames Research Center [6]. For demonstration purposes, a STOVL aircraft model, which is described below, was chosen with its associated HUD symbology, HDD instrumentation, and cockpit effector configuration.

The HUD symbology was generated and updated on the visual system development station. The HUD symbology is projected, in addition to the scenery, on the forward projection screen. The graphics software libraries were provided with the visual system to aid in the creation and implementation of HUD symbology, especially the generation of various fonts and colors. Graphics routines for all the displays were written in the C computer language. The displays and scenery were modified to reflect an integrated engine and airframe control task, typical of a STOVL aircraft. Figure 3 shows an example HUD symbology which was implemented on the flight simulator. The symbology includes a pitch ladder, heading scale, aircraft reference symbol, and flight path symbol. Additionally, engine and aircraft parameters such as altitude, velocity, and nozzle angle also are displayed.

The HDD instrumentation was displayed on a touch sensitive, 19 inch, color monitor, and was generated using the Pepper Graphics NNIOS Development Software. Figure 4 shows an example HDD instrumentation panel that was implemented on the flight simulator. For this STOVL aircraft example, an altimeter, compass, horizontal situation indicator, and airspeed indicator display the airframe parameters in real time. Engine parameters are displayed on fan speed, fuel flow, nozzle angle, and nozzle area gauges. These parameters were chosen for display to aid the control design engineer in evaluating the engine dynamics. Additional engine parameters can be displayed on these simulated gauges. Flight mode information is displayed and altered through the touch sensitive screen. A keypad function allows the user to change the mode information or to select a new scenario or starting point for the simulation.

The switches and effectors in the mock-up fighter cockpit are implemented to reflect the simulation of an integrated flight and propulsion control task. The cockpit effectors were based upon a "rate system" command structure. This rate system was implemented to accommodate the three modes of flight that the example STOVL aircraft can encounter: cruise, transition, and hover. With the rate system commands, the longitudinal stick provides pitch rate/attitude hold; the lateral stick provides roll rate/bank angle hold; the rudder pedals provided sideslip commands; and the linear throttle commands forward velocity. An additional analog effector was added for this simulation -- a rotating thumbwheel. The thumbwheel, positioned on the linear throttle, commands flight path angle during the simulation. A diagram of the cockpit effectors and their functionality is found in Figure 5.

SIMULATION TESTING

In order to test and evaluate the hardware and software capabilities of the flight simulation facility, an example control design, aircraft model, and engine model were established. The vehicle model for this simulation test is a six degree of freedom, delta winged E7-D aircraft with a multi-nozzle

turbofan engine, which is shown in Figure 6. The airframe is configured with an ejector nozzle, a ventral nozzle, a 2D-CD aft nozzle, and a Reaction Control System (RCS). The RCS allows for control of aircraft attitude during hovering flight. The engine for this aircraft is a mixed flow, vectored-thrust configuration. Further information about the vehicle, the airframe model, and the engine model can be found in reference [7].

Figure 7 shows the discrete linear control design that is executed on the control computer. The pilot inputs from the cockpit effectors are sent to the control computer. These signals are scaled by the input effector gradients. The signals are also filtered by the linear ideal response models. These ideal response models convert the pilot selections of acceleration, pitch rate, flight path angle, roll rate, and sideslip, into desired velocities, accelerations, and body angles or rates for the controller. The ideal response models are based upon desired handling quality characteristics of the E7-D aircraft, response dynamics of the airframe and engine, modal coupling and decoupling, and flight kinematics for turning flight consistency. [8]

The output from the ideal response models is the desired perturbation of the airframe outputs. This desired amount of change is adjusted by the measured outputs from the simulated aircraft dynamic models. This desired amount minus measured amount is input to the discrete linear control, $K(z)$. For this example, the integrated flight and propulsion controller is a reduced order H-infinity design, [8] which is a linear, 21st order system. The output from the control is then added to the nominal plant actuator values and sent to the airframe and engine physics models located on the simulation computer.

Figure 8 shows the calculations performed by the simulation computer of the linear E7-D airframe and engine perturbation models. The system inputs from the control computer minus the nominal physics model inputs are fed into the linear plant, $G(z)$. The basic integrated engine and airframe models are a 14th order system with 12 inputs and 10 outputs. The output from the engine and

airframe plant is the measured amount of change from the dynamic models. This output is added to the nominal plant outputs and fed back to the control computer and to the coordinate transforms. Coordinate transformations are performed to provide body reference system airframe and engine parameters to the visual system for visual scenery and display updates.

During integration and testing of the facility, it was desirable to obtain a measurement of the computational capability of the simulator, therefore, several benchmarks were performed. Since the simulation facility consists of several different computer systems connected by various communication links, one important benchmark was to insure correct timing between the distributed portions of the simulation running on these separate computer systems.

Figure 9 shows a timing diagram for the cockpit, the control computer, the simulation computer, and the visual generation system along with the handshaking that occurs between the systems. The control computer runs asynchronously from the other computer systems. The integrated control algorithms execute every 40 milliseconds using timer generated interrupts. The first event to occur within the control computer upon an interrupt is the I/O with the cockpit and the simulation computer. The controls microcomputer then executes the integrated airframe and propulsion control algorithm. The MINDS data acquisition system runs in the spare time. The simulation computer updates the airframe and propulsion models every 20 milliseconds. Upon each update it performs the analog I/O with the controls computer and then executes the airframe and engine models. Every 40 milliseconds the simulation computer sends updated visual information digitally to the visual generation system over the DR11W interface. The visual generation system update is thus synchronized with the simulation computer. After reading the inputs, the visual system will update the cockpit HDD, the HUD, and the scenery. It then waits for new data to be provided from the simulation computer.

CONCLUSIONS

The integrated propulsion and flight simulator has successfully been designed, built, and demonstrated as a real time, pilot-in-the-loop, evaluation station for integrated engine and airframe control laws. Benchmark tests for timing and integration show that the flight simulation system performed in real time, without degradation of the visual displays, for a sample STOVL aircraft application. For its designated use as an engineering evaluation simulator, the flight simulator system performs well for real time display of flight control parameters and engine control parameters. The simulation system at NASA Lewis, as further confirmation of its real time capabilities, will undergo engineering evaluations of the system configuration to evaluate interface structures and control mode structures. Additionally, the simulator will be used to evaluate nonlinear, integrated control designs.

A planned improvement to the flight simulator facility is a real time digital communications network between facility components. Networking hardware and software will expand the capability to monitor simulation and control parameters of all hardware in the simulation in real time. This expansion will also allow for easy reconfiguration of the system components for use with actual engine hardware in the loop. Finally, through the modular system design of the flight simulator, component upgrades and modifications can be made to accommodate future research, or expand to motion based simulation.

ACKNOWLEDGEMENT

The authors wish to thank Dr. Duane Mattern for his assistance on the control design portion of this paper, and Mr. John DeLaat for his technical assistance.

REFERENCES

[1] Paragon Graphics Visual System User Manuals, Paragon Graphics, Inc. Orlando, FL.

[2] Delaat, J.C., Soeder, J.F., "Design of a Microprocessor-Based Control, Interface, and Monitoring (CIM) Unit for Turbine Engine Control Research", NASA TM-83433, 1983.

[3] Version 7.1 ADSIM/ADRIO Manuals. Applied Dynamics, Inc. Ann Arbor, MI.

[4] PC11 Users Guide. Omnicorp Graphics Corp. Houston, TX.

[5] MATRIXx/Autocode User Manuals. Integrated Systems, Inc., Santa Clara, CA.

[6] Merrick, V.K., Farris, G.G., Vanags, A.A., "A Head Up Display for Application to V/STOL Aircraft Approach and Landing", NASA TM-102216, 1990.

[7] Akhter, M.M., Vincent, J.H., Berg, D.F., Bodden, D.S., "Simulation Development for US/Canada Controls Technology Program", Proceedings of the Twentieth Annual Modeling and Simulation Conference, University of Pittsburgh, Pittsburgh, PA. June 1989.

[8] Garg, S., Mattern, D.L., Bright, M.M., Ouzts, P.J., "H-Infinity Based Integrated Flight Propulsion Control Design for a STOVL Aircraft in Transition Flight", NASA TM-103198 and Proceedings of the AIAA Guidance, Navigation and Control Conference, Portland, Oregon, August 20-22, 1990.
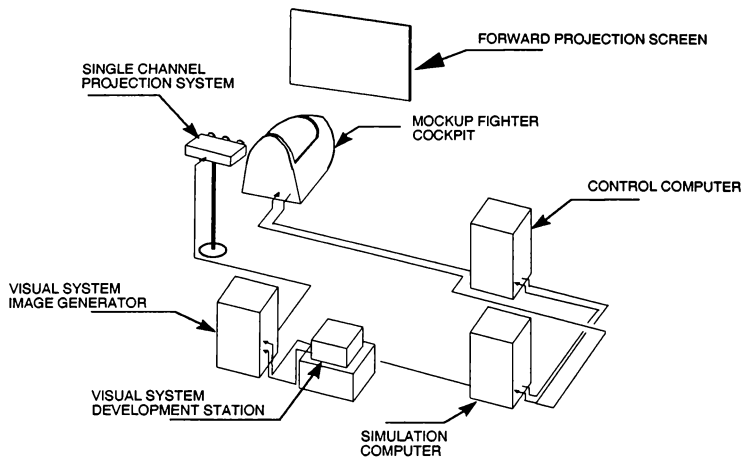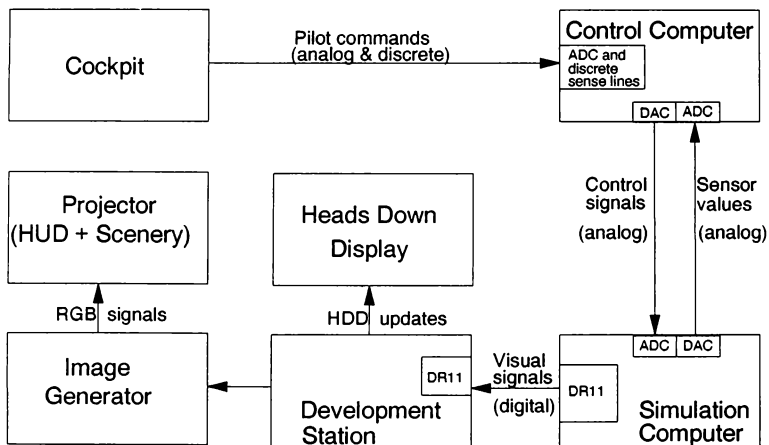
Figure 1. Flight Simulator Facility
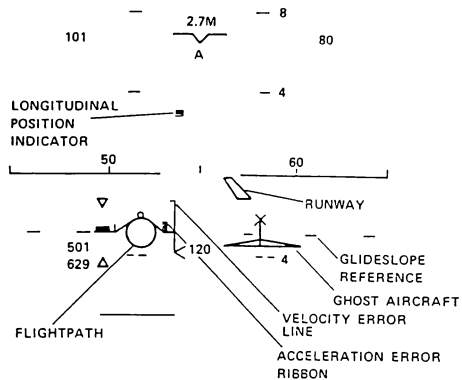

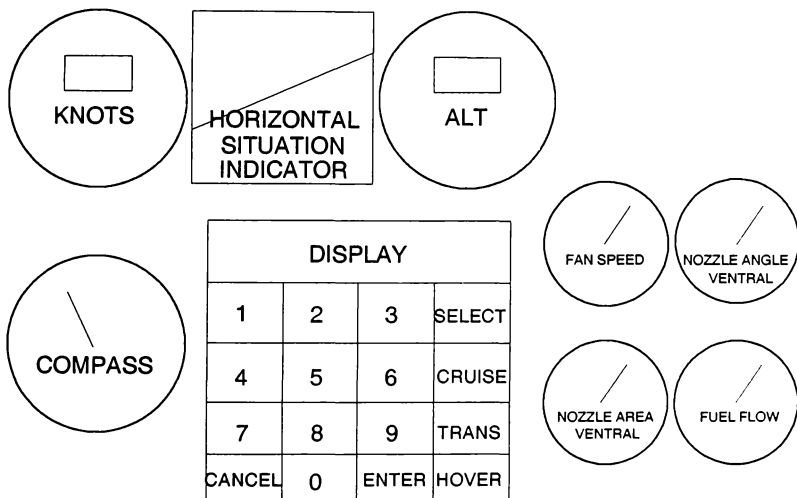Figure 2. System Communications

Figure 3. Heads Up Display Format [6]

Figure 4. Heads Down Display Format

H = hover T = transition C = cruise

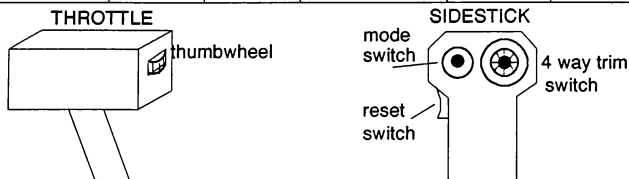| pilot command \ control inceptor	Throttle	Thumb Wheel	Longitudinal Stick	Lateral Stick	Rudder Pedals
accel/decel		H C T			
flightpath	T				
roll/rollrate				C T	
pitch/pitchrate			C T H		
sideslip					C T H
velocity	C			H	
altitude rate	H				

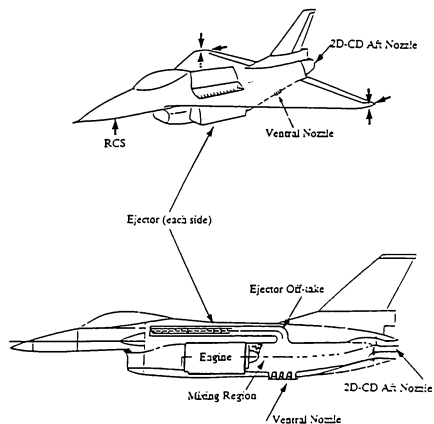Figure 5. Control Modes in Cockpit

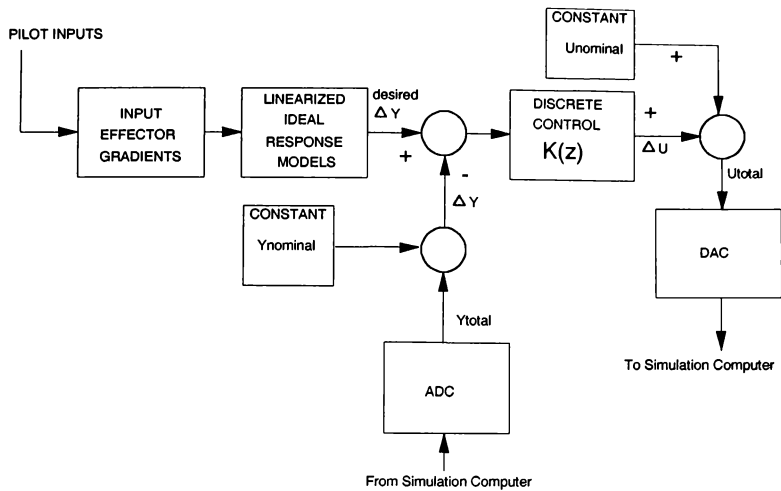Figure 6. E7-D Configuration Aircraft

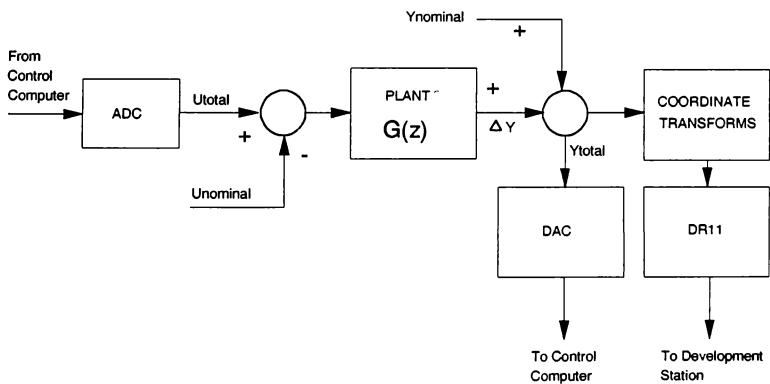Figure 7. Integrated Control Design on Control Computer

Figure 8. Integrated Plant on Simulation Computer

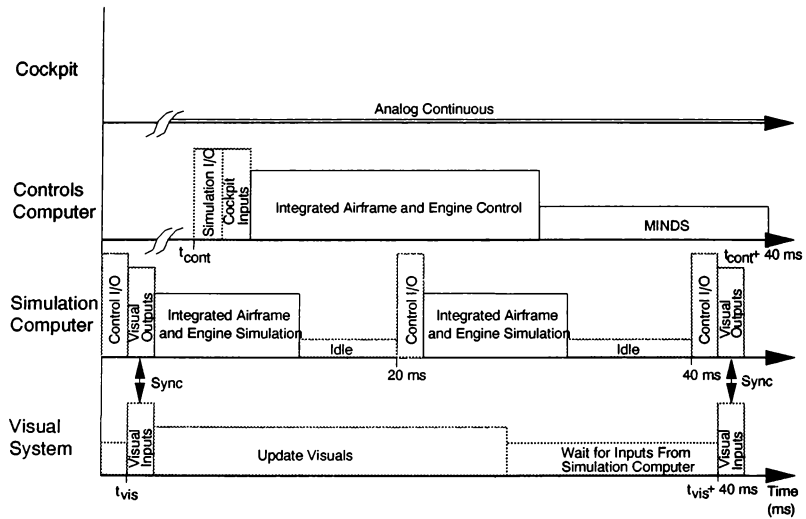Figure 9. Timing Diagram of Computers

High Performance Computing System for Flight Simulation at NASA Langley

Jeff I. Cleveland II*, Steven J. Sudik, and Randall D. Grove
NASA Langley Research Center
Hampton, Virginia 23665-5225

NASA's Langley Research Center (LaRC) has been utilizing real-time flight simulation to support aerodynamic, space, and hardware research for over fifty years. In the mid-1960s LaRC pioneered the first practical real-time digital flight simulation system utilizing Control Data Corporation (CDC) 6600 computers. In 1976, the 6600 computers were replaced with CDC CYBER 175 computers. In 1987, the analog-based simulation input/output system was replaced with a high performance fiber optic-based digital network. With the increased complexity and higher performance requirements for the simulation of modern aircraft, LaRC is replacing the CDC computers with Convex Computer Corporation supercomputers. This paper reviews the current hardware and software, experience with the new system, and current status and plans.

INTRODUCTION

An unpublished survey of flight simulation users at the NASA Langley Research Center (LaRC) conducted in 1987 projected that computing power requirements would increase by a factor of eight over the coming five-years (Figure 1). Although general growth was indicated, the pacing discipline was the design testing of high performance fighter aircraft. Factors influencing growth included: 1) active control of increased flexibility, 2) less static stability requiring more complex automatic attitude control and augmentation, 3) more complex avionics, 4) more sophisticated weapons systems, and 5) the need to simulate multiple aircraft interaction, the so called "n on m" problems. This requirement for more computing power is, if not industry wide, at least common to the fighter aircraft segment.

In use since 1976, two single processor Control Data Corporation CYBER 175 computers, tightly coupled through extended memory, are used to support flight simulation. In 1987, a new digital data distribution and signal conversion system, referred to as the Advanced Real-Time Simulation System (ARTSS), was put into service at LaRC. This system, using the Computer Automated Measurement and Control (CAMAC) technology, replaced two twenty year old analog-based systems. The ARTSS has been very successful and is described in references 1 through 6.

Having decided to continue using centralized computers, LaRC issued a Request for Proposals in May, 1989 and subsequently awarded a contract to Convex Computer Corporation in December of that year. The resulting computational facility provided by this contract is the Flight Simulation Computing System (FSCS). This paper presents the hardware and software comprising the FSCS, experience to date, planned implementation, and current status and plans.

NEW SYSTEM

Hardware

Computer hardware is being acquired in a phased installation. In June, 1990, a Convex C220 computer (expandable) was installed as the first interim computer. One CAMAC interface was included so that one simulation application could be accommodated with this configuration. In March, 1991 a Convex C220 computer was added to enable testing and transition to the first release of the real-time operating system to be delivered later in the year. An additional CAMAC interface was included to allow either one simulation application on each of the two computers, or with transfer of interface hardware and CPU, two simulation applications on the initial C220 expanded computer. An advanced technology Convex computer is scheduled for delivery in September, 1991. Once the advanced technology computer has been integrated into the ARTSS system, the two C220s will be phased out.

A second advanced technology Convex computer is planned for the fall of 1992. Additional contract options provide for the acquisition of additional CPUs

---

* Senior Member AIAA

and CAMAC interfaces. A real-time application will be able to utilize multiple CPUs. The final advanced technology hardware configuration is shown in Figure 2.

Operating System

The initial operating system delivered with the first Convex C220 is a modification of the standard Convex UNIX operating system. The modification allows a real-time application to take exclusive use of a CPU and allows the application to pre-page memory and lock all pages into memory so that no page faults will occur. With this operating system, a real-time application runs on one CPU in real-time while the remaining CPUs run general UNIX programming. A recently installed extension allows multiple real-time applications to run simultaneously in real-time, each application using its own dedicated CPU.

The real-time operating system being developed will incorporate a real-time kernel that is the base level of the operating system. This kernel will support all of the real-time functions including exclusive use of a CPU, pre-paged locked down memory, deterministic interrupt response, and real-time input/output support. The first version will be a run-time only operating system. Application development and other UNIX tasks will be accommodated on another computer linked to the real-time computer via TCP/IP. The final operating system will build UNIX as a "real-time" task controlled by the real-time kernel. With this operating system, real-time applications can operate in real-time simultaneously with program development in UNIX using the same CPU.

The operating system will include a special application real-time debugger. This debugger is based on the GNU *gdb* debugger program and includes extensions for control and special displays for real-time.

Language and other factors

Currently, almost all simulation application programs are written in the FORTRAN language. Since the development of real-time digital simulation at LaRC in the mid-1960s, applications have been developed on computers with extended precision hardware. Most of the simulation applications will be converted to the FSCS using 64-bit word length to maintain the model accuracy achieved with previous systems. In addition to FORTRAN, some portions of applications are being developed in the C and Ada languages.

The real-time kernel portion of the operating system is written in C. Unlike the previous system where the input/output driver codes were written exclusively in assembly language, the new system allows input/output drivers to be written in C and offers debugging capability for these drivers.

LaRC IMPLEMENTATION

CAMAC Software Driver

Most of the driver software for the CAMAC highway executes in the I/O processor (known as a VIOP (VMEbus I/O Processor)). During other than synchronized real-time, driver code executes in the CPU to activate the VIOP portion. For the real-time portion of the application's execution, all highway I/O is accomplished by direct interaction between the VIOP and the application. The real-time application has exclusive access to its highway and the VIOP that drives the highway. The dedicated VIOP is not interrupt driven; rather, it polls request areas in the application and samples various hardware elements to determine the occurrence of external events. With very short polling loops, polling meets the performance required for LaRC real-time operation.

The VIOP is a Motorola 68020 processor with its own operating system known as EGOS (Event Governed Operating System). EGOS is used on both the interim system and the final real-time system so little modification is needed to the VIOP driver software during migration. On the final real-time system, the VIOP driver will be a task and therefore will be more easily reloadable than on the interim system.

In real-time operation, the application receives input, does computation, and transmits output in a very regular fashion. An input-compute-output pass is referred to as a real-time frame. All frames are exactly the same length as determined by real-time clock signals that the VIOP senses. To the VIOP, the real-time frame is composed of several operations:
1. It waits for and processes the clock signal.
2. It transfers synchronized input to the application.
3. It signals the application to start processing.
4. It scans the hardware for asynchronous I/O signals.
5. It scans the application for asynchronous I/O requests.
6. It waits for the application to signal that synchronized output is ready.

Time remaining in the frame permitting, it can then do more of steps four and five. And finally, it returns to the start of the loop for the clock signal of the next frame.

On the interim system, the VIOP accesses registers known as the Time-of-Century clock to determine when the application has taken too long to signal clocked output. This condition is diagnosed as an error that may cause the application to abort. On the FSCS real-time system, there will be an additional real-time background capability. The application will

be allowed to execute code that is not time-critical during real-time input/output periods and during the compute period when the time-critical portion has completed its computations. The VIOP will interrupt this background code at the end of input to give control to the time-critical portion of the application. At the end of the time-critical code, the application can resume where it was interrupted in the background code.

Configuration Management Software

Maintenance of site configuration information is done in two phases by two utilities: Site Compiler (SC) which describes in detail individual sites and Site Linker (SL) which groups individual site descriptions together into configurations. A simulation application typically uses one simulation control console, one or more simulator sites, and often one or more sites that have a graphics computer. Hardware in the consoles is always configured the same; but different applications may use varying subsets of hardware at the simulator sites. SC builds intermediate files describing each console and each unique use of the other sites. SL generates configuration files by combining information in selected SC output files that will be used by a given application.

Functionally SC is much more complex. It accepts input directives describing the total hardware available at each site and other input directives describing the various subsets of that hardware that may by used by applications. It generates fairly complex binary output files containing the commands needed to drive the hardware as configured and containing data used by Real-Time Supervisor at run time.

SL is functionally fairly simple. It collects and groups together from various site files, initialization operations, site names, and timing information. For each configuration, it generates a binary file containing information for Real-Time Supervisor and a binary file of information for the software driver. SL also generates a listable report describing each configuration for the application programmer. This report includes block lengths, logical names, device types, and other information.

Real-Time Supervisor

Real-Time Supervisor is a suite of routines written in C, FORTRAN and assembler that resides in a run-time library used by the application programs. It provides most of the job's interactions with the system and the real-time environment. It has three main classes of functions. First, it interacts with the sites on the highway. Second, it drives the real-time recording device (DRR). Third, it interacts with the

simulation operator through the terminal and performs error recovery processing.

The highway portions of Supervisor define communication areas and buffers, make asynchronous requests for input or output, retrieve and reformat this data, process and reformat the synchronous I/O, and fetch up-line quasi-interrupts called demand messages. This function is basically identical on the initial system and the final real-time system.

The DRR (data recording and retrieval) portion defines buffers and variable names for recording data, performs input/output to and repositions DRR files. The interim system implements this capability in memory with a spooler that periodically records it on disk. The final real-time system has a native real-time disk capability.

The terminal I/O and error recovery portions are the most complicated. One of the main aspects of the LaRC real-time design philosophy is that the application does not terminate unless the user explicitly commands it to. The RTERROR routine recovers from every error it can and displays a menu giving the user several options. He can recover if he has predefined a recovery address, he can display and analyze portions of code and data, and he can dump memory. The hardware architecture of the FSCS makes implementation of error recovery more difficult than on previous systems. On previous systems, the context of an application was small and easy to localize, so restarting an application at a different place was relatively simple. The existence of stack frames on the FSCS make this more difficult. The strategy is to allocate different areas on the stack for different contexts. Because of this limitation, only error recovery and frame overrun recovery were implemented on the interim system. The foreground/background capability was not implemented but will be implemented on the final real-time system. On the final real-time system, sub-tasks and light-weight contexts may allow easier recovery and context switching. The Convex architecture has also hampered implementation of the capability to restart a previously saved memory image.

ARTSS Hardware Modifications

To support the FSCS, some modifications to the ARTSS are required. To support additional computers during the transition period, additional RS-232 ports for the configuration switch controller are required. To retire obsolete equipment and to provide additional ports, a UNIX based 386 class microprocessor system has been developed as a replacement for the switch controller. The new controller system offers all the functionality of the old controller, adds additional operator capabilities for remote

controllability, and is much easier to maintain and update.

With the additional requirement to support secure and non-secure processing simultaneously, an additional real-time clock is required for the secure facility. To support the higher frame rates possible with the new FSCS, a smaller basic time interval is required to form the simulation frame rate. A new simulation real-time clock has been designed and is being implemented at LaRC (reference 7).

In the previous paper (reference 4) concerns about the transport delay caused by real-time input/output data formatting were presented. Experience has shown that transport delays caused by CPU data formatting to accommodate real-time input/output degrade real-time performance. A new VME-based CAMAC serial highway driver meeting LaRC requirements is being developed by KineticSystems Corporation as a standard product. This new hardware requires no data formatting for real-time input/output. In addition, input/output latency is further decreased by improved hardware interface to the software driver.

SIMULATION APPLICATIONS

The real-time facility supports over forty simulation activities with one-third in production during a week's operation. These simulation projects span the research field from general aviation aircraft to hypersonic aerospacecraft. Real-time simulation applications use one of seven simulators in their research effort, as well as graphic computers for the cockpit displays, computer generated imagery for viewing the outside world, and mini/micro-computers for other special effects.

The Transport Systems Research Vehicle (TSRV) simulator is a cockpit of a Boeing 737 aircraft using a highly modified flight deck. TSRV studies include investigations of flight crew requirements and operational procedures for a ground-to-air data link air traffic control (ATC) scenarios. Another TSRV application is the development of on-board graphical weather interactive displays from ATC weather information sources.

To support simulations using the TSRV simulator, the FSCS with its increased CPU power and its increased memory capacity will enable expanded modelling in the areas of guidance algorithms, navigation information, flight path algorithms, and data link algorithms.

The Differential Maneuvering Simulator (DMS) consists of dual 40-foot projection spheres with cockpits that have out-the-window views, heads-up and other cockpit displays, and other-aircraft target projection systems. The DMS is used to investigate the flight scenarios of modern and prototype high-speed jet aircraft in single pilot, one-on-one, or two-versus-one modes of operation. Using the single pilot mode, NASA and the U.S. Navy are conducting high angle-of-attack simulation studies to develop guidelines for nose-down control requirements. Single air combat simulations incorporate maneuvering logic programs to drive the target aircraft in evasive actions. The other two modes of operation, one-on-one and two-versus-one, aid in developing combat tactics and strategies. New applications are investigating helmet mounted displays for the presentation of aircraft related information and visual displays.

The new FSCS offers expanded capabilities for high performance aircraft modelling. Increased memory capacity permits databases to include all the data required for missions (full mission scenarios), from leaving the hangar, through take off and landing and the flight objective. Increased computational speed coupled with increased memory permit the use of more sophisticated models.

The General Aviation (GA) simulator consists of a general aviation cockpit with out-the-window visual displays and a limited motion capability. Simulations using the GA simulator include an engine out landing study, handling qualities in turbulent weather study, and a highway-in-the-sky take off and landing study for novice pilots.

The Visual Motion Simulator (VMS) is a six-degree-of-freedom motion base platform with a dual seat cockpit fitted with out-the-window visual displays. Studies using the VMS include the simulation of transport aircraft encountering wind shears from microbursts and high speed civil transport investigations of mach 3 passenger travel over ranges of 6500 nautical miles. The hypersonic aerospacecraft research studies a powered vehicle designed to takeoff from the ground, accelerate to hypersonic speed, enter earth orbit, re-enter the earth's atmosphere, decelerate and land at an airport. Another area of study is the Personnel Launching System (PLS). The PLS vehicle will be used to ferry personnel to and from a space station or other vehicle in space. This system will have a flight envelope similar to the space shuttle with wide range of mach number, altitudes from ground to space, and flight/space dynamic models.

The FSCS expands the capabilities for full mission scenarios for high speed civil transport simulation, hypersonic aerospacecraft simulation, and PLS simulation. With the increase in memory capacity, the FSCS permits high fidelity simulations of weather models including models containing microbursts.

Hardware-in-the-loop simulations include space station structures that are large flexible units with numerous excitation modes. System identification

and control law testing is being done with the hardware and sensors in closed-loop and open-loop operation. The active flexible wing simulation (Figure 3) consists of the math model on the mainframe computer and the control algorithms on an external control computer in a closed loop.

The FSCS has the power required to support simulation of flexible space structures at the required rate of 1000 frames per second. The higher frame rate supports the analysis of the desired solution frequencies while the increased power supports additional excitation modes. In combination with the interfacing capabilities of ARTSS, the FSCS is able to support many hardware-in-the-loop simulations.

STATUS AND PLANS

The two interim computers have been installed and support simulation research production. The second delivery of the interim operating system, supporting multiple real-time applications, has been installed and verified. Installation of the first version of the real-time operating system is planned for July, 1991 with acceptance of the full real-time operating system planned for January, 1992.

The simulation application programs are being converted to the new flight simulation computing system. An initial set of applications are being converted to the interim computer to gain experience with the new system. Two real-time networks are currently available, one for limited real-time production and one principally for system testing. A new DMS secure application is being programmed to run on the FSCS computer in order to test the operational procedures for the new secure room facilities. Selected other new application programs will be directed to the FSCS to reduce the generation of programs for both the old and new systems.

The initial FSCS system with its new hardware and software requires little simulation program modifications. As the system software approaches maturity near the end of 1991 more applications will be shifted to the FSCS computer and its two real-time highway networks. The TSRV conversion is the last major phase and the most difficult since most of the TSRV applications require two CYBER computers to operate the simulations. The conversion effort is scheduled to be completed in mid 1992.

CONCLUSION

NASA Langley Research is at the mid-point in the development of a computing system to simulate in real-time increasingly complex and high performance modern aircraft. Utilizing centralized supercomputers coupled with a proven real-time network technology, scientists and engineers are performing advanced research using flight simulation. Completion of this system will provide high performance flight simulation meeting requirements into the late 1990s.

REFERENCES

1. Crawford, D. J. and Cleveland, J. I. II, "The New Langley Research Center Advanced Real-Time Simulation (ARTS) System," AIAA Paper 86-2680, October 1986.

2. Crawford, D. J. and Cleveland, J. I. II, "The Langley Advanced Real-Time Simulation (ARTS) System," AIAA Journal of Aircraft, Vol 25, No. 2, February 1988, pp. 170-177.

3. Crawford, D. J., Cleveland, J. I. II, and Staib, R. O., "The Langley Advanced Real-Time Simulation (ARTS) System Status Report," AIAA Paper 88-4595-CP, September 1988.

4. Cleveland, J. I. II, Sudik, S. J., and Crawford, D. J., "High Performance Processors for Real-Time Flight Simulation," AIAA Paper 90-3140-CP, September 1990.

5. Cleary, R. T., "Enhanced CAMAC Serial Highway System," presented at the IEEE Nuclear Science Symposium, San Francisco. California. October 23-25, 1985.

6. ANSI/IEEE Standards 583, 595, and 675, Institute of Electrical and Electronic Engineers, 1976.

7. Bennington, D. R., "Real-Time Simulation Clock," LAR-13615, NASA Tech Briefs, June 1987.

BIOGRAPHY

Jeff I. Cleveland II is the Project Engineer of the LaRC Flight Simulation Computing System. He received a B.S. in Electrical Engineering in 1963 from Texas A&I University and an M.S. in Electrical Engineering and Computer Science in 1970 from The George Washington University. He has worked in the field of flight simulation and operating system software since 1963.

Steven J. Sudik is a Project Analyst with Unisys Corporation. He is the group leader for the Flight Simulation Computing System software development team. He received an A.B. in English and Philosophy from Marquette University in 1966 and an M.A. in Philosophy from Indiana University in 1971. Prior to coming to LaRC he held positions as a systems programmer and business applications programmer. He has worked at LaRC since 1975 in operating systems support.

Randall D. Grove is the Lead Simulation Engineer for the LaRC Flight Simulation Computing System. He received a B.S. in Applied Mathematics in 1963 from the California State Polytechnic College at San Luis Obispo. He has worked in the field of flight simulation and application program development since 1963 at LaRC.

TRADEMARKS

UNIX is a trademark of American Telephone and Telegraph.

CDC and CYBER are trademarks of Control Data Corporation.

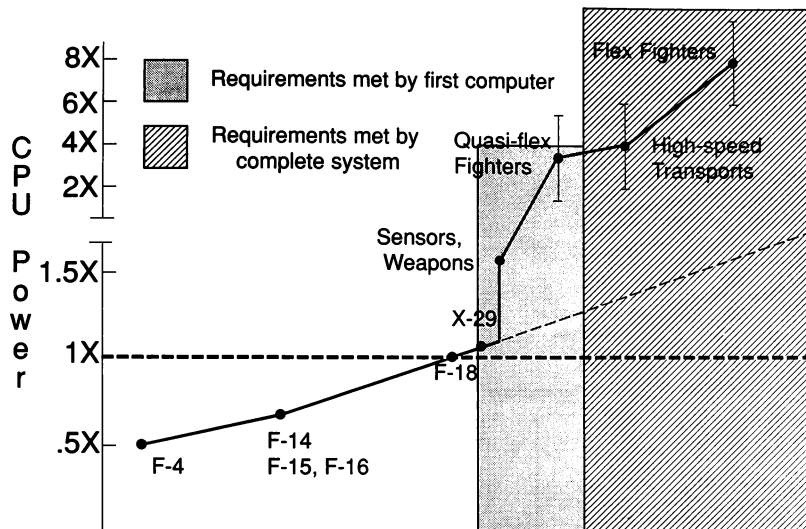Convex is a trademark of Convex Computer Corporation.

Figure 1

NASA Langley Research Center CPU power requirements for flight simulation

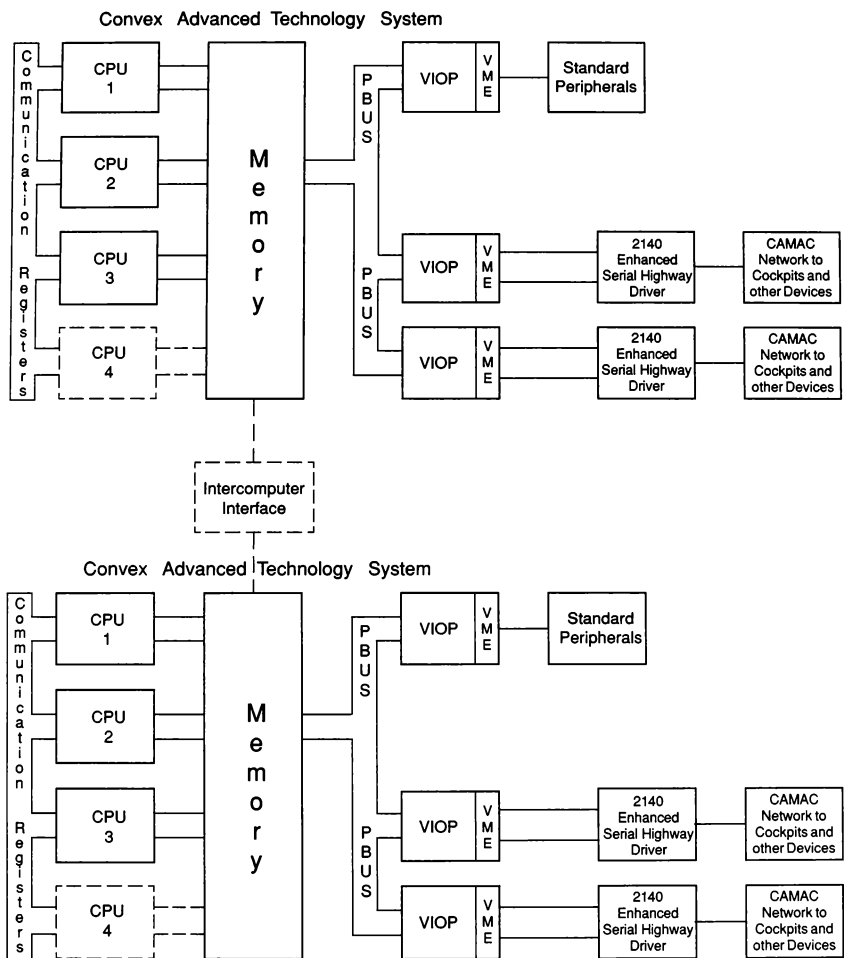Convex Advanced Technology System

Figure 2

Final System Configuration

LLEO 0.00 LLEI 0.00 RLEI 0.00 RLEO 0.00
LTEO 0.00 LTEI 0.30 RTEI -0.30 RTEO 0.00

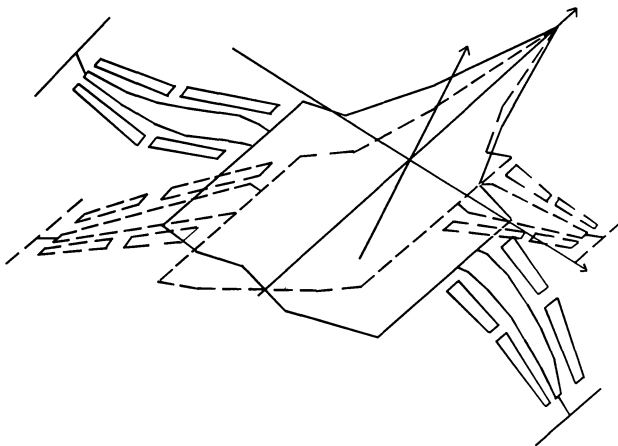Q 350.00   MACH   0.90   CCFR 8.00
MAG 1.00

Figure 3

Aircraft display pitch, roll, and yaw angles, control surface deflections,
and model deformation (magnified)

A NEW RADAR SIMULATOR ARCHITECTURE

Roderic C. Deyo
Kenneth C. George
John T. Mason

Evans & Sutherland Computer Corporation
Salt Lake City, Utah

ABSTRACT

A new real-time object-oriented software-based approach to multi-mode radar simulation utilizing a scaleable pipeline/parallel multi-processor architecture and a standard data bus has been developed by Evans & Sutherland. This system takes advantage of special analytic feature primitives which, when combined with specialized processing, provide an extremely rich, accurate, and correlated image. This paper explores the functional requirements for various radar modes and effects, summarizes how the pipeline/parallel hardware and software architectures meet these requirements, and presents examples of the processing flow associated with the representative ground-map radar mode. The paper addresses how modeling strategies affect correlation with visual simulation and how the tradeoffs associated with the choice of different types of feature primitives available in the run-time data base determine ultimate system capacity and performance. The novel feature primitives employed in the system are linked to improving radar database feature density, level-of-detail management, visual correlation, and anti-aliasing within the context of a parallel system architecture.

INTRODUCTION

Radar image generation is required to meet a variety of needs in modern flight simulation, including crew training, mission training and rehearsal, and engineering evaluation and development. A multi-mode radar simulator must be designed with simulation fidelity and flexibility in mind. It must be able to operate in real time, handling realistic aircraft motion and the full tactical environment, while providing faithful radar images and updates. Each simulated radar mode must provide the full spectrum of unique effects as well as reproduce all the detail necessary to meet mission training requirements. In particular, the radar simulator must be able to render a rich, high-density database. Finally, the radar simulator must operate in an integrated training environment and provide optimum correlation with other simulated sensor channels as well as out-the-window visual images.

To provide its customers such capabilities at a reasonable cost, Evans & Sutherland has developed a radar simulation system, the E&S Radar Image Generator (ESRIG), based upon a scaleable pipeline-parallel multi-processor architecture using an industry standard communication and data bus. The ESRIG hardware takes advantage of recent developments in digital signal processor chip technology and a carefully chosen memory architecture to provide the processing power required to support a software-based approach to radar simulation. The hardware is powerful, yet relatively simple, and when combined with programming in the standard ANSI C language allows the complete system to be modular and easily supportable. The system incorporates a real-time object-oriented software architecture which allows it to be readily tailored and tuned to meet current and future specific operational requirements for different types of radars without significant hardware modification. A flexible display subsystem supports custom tailoring of display formats, annotation, and symbology. The ESRIG can readily be scaled to provide additional capabilities without software modifications by varying the number of processors. Finally, the ESRIG was designed to effectively employ specialized radar database primitives. These primitives provide the rich collection of data objects and attributes needed to render the high-density geo-specific correlated images required for mission training.

This paper describes in general terms the Evans & Sutherland radar simulator system architecture and addresses how the hardware, software, and database designs have been optimized to meet current and future functional requirements. To do so, we concentrate on the fundamental real-beam ground map radar mode. We conclude by discussing the very import topic of correlation with other sensor simulators.

ESRIG FUNCTIONAL REQUIREMENTS

A careful examination of recent radar simulator specifications for several current and planned airborne radar systems provides the basis for determining a set of typical operational and functional requirements for a multi-mode radar simulator. These radar simulation requirements can be grouped as follows: real-beam and Doppler ground map radar (GMR), air-to-air (AA) search and track modes, and air-to-ground (AG) target tracking and ranging modes. This set of modes covers most of the operational capabilities of the primary airborne radar set. Auxiliary airborne radar systems must also be simulated, including terrain following radar (TFR) and radar altimeter (RALT). Additionally, radar simulators are often required to provide ancillary functions, such as occultation or intervisibility processing. Such calculations, while not directly a part of airborne radar systems, are required to support tactical environment simulation. The ESRIG modes and functions must be closely correlated to avoid negative training cues.

In general, to ensure tactical fidelity, a complete radar simulation must provide relevant radar returns and effects from terrain, features, weather, targets, and counter-measures. Multipath and beam effects, such as far-shore and leading-edge brightening, must be given special attention. In addition, each mode must also provide a complete pilot/operator interface and display, including proper screen updates and symbology.

Finally, and most important, when the ESRIG is part of a multi-sensor simulation environment, adequate correlation between sensor simulators is crucial in providing an effective training experience.

The ESRIG has been carefully designed to meet these and other radar simulation requirements. In discussing its architecture, a key mode to consider is real-beam GMR since its implementation encompasses a majority of the techniques required by other modes.

GMR Functions

A real-beam ground map radar provides a two-dimensional map-like polar display of returned energy from terrain and features within the chosen range and antenna azimuth scan pattern. A series of pulses are transmitted in the current antenna direction and the time-resolved returns are used to generate a vector of intensities (called a sweepline). The real-beam display, or radar scope, is usually formatted as a circular or sector scan (called a plan position indicator, or PPI) matching the range and azimuth footprint selected by the operator. The area to be mapped is chosen by adjusting the range scale and sector width and offset controls in order to produce an image concentrated on some region of interest. Typical airborne GMR sets have range scales of 2 to 80 nautical miles and sector widths of 10 to 120 degrees. Antenna sweep rates of 60-120 degrees per second are commonly employed.

When simulating real-beam GMR mode, the ESRIG provides the following essential functions:

- Range scale selection
- Antenna sweep and sector control
- Radar equation processing
- Real-beam radar effects simulation
- Range and azimuthal antenna filtering
- Polar-to-raster scan conversion

In addition to real-beam radar effects, the ESRIG accurately simulates radar returns and effects due to weather, moving air and ground targets, and the tactical environment (including chaff and jamming).

The following system functions allow the ESRIG to function effectively in an integrated real-time training environment:

- Database management and paging
- Host interface communications
- System startup and initialization

- Operator control interface
- Error handling
- System diagnostics

Other ESRIG Modes

In general, the GMR mode functionality provides a foundation for other modes and submodes within the ESRIG system. For example, the processing flow for real-beam expand, stabilized, and Doppler-enhanced ground map modes is based on real-beam GMR simulation. However, the ESRIG AA modes and AG target tracking and ranging modes of the radar furnish a larger amount of information about a smaller number of individual objects than does GMR mode. The tracking modes of a real radar furnish automatic continuous measurements of range, angle, and/or Doppler frequency for a specified target return. Target tracking and ranging modes are somewhat simplified in the simulated environment since all target state information is readily available. Terrain and feature information can be employed to estimate clutter and consequent target lock-on probabilities.

The airborne TFR scans a narrow (approximately 10-degree) azimuthal sector of terrain ahead of the aircraft and provides range and elevation data for terrain and obstacles to a separate radar display. Data is also provided to the main computer in order to generate autopilot pull-up or push-over commands during terrain-following flight. The ESRIG simulation of this mode uses essentially the same methods as GMR, but adding the specialized TFR display.

An airborne RALT consists of a separate radar antenna and transmitter-receiver combination which measures the distance from the aircraft to the nearest obstacle that produces a return above a given threshold in a vertical upright cone beneath the aircraft. In the ESRIG RALT simulation, special care is taken so small but significant vertical features are not to be missed in the database. Simply calculating aircraft nadir height may not always meet mission training requirements.

In addition to simulating the operation of on-board radar equipment modes, the ESRIG can also perform other calculations related to the tactical training environment. This environment includes electronic emissions from both friendly and enemy ground and airborne radars and electronic countermeasures such as chaff and jammers. To assess the tactical impact of these potential threats, the simulation system needs real-time information on threat radars which may be illuminating the simulated aircraft. This requires that the ESRIG must track a set of up to several hundred potential emitters and perform an intervisibility calculation for each to determine which threats are in shadow with respect to the aircraft, which threats are oriented such that they can illuminate the aircraft, etc. Consistency of these calculations and correlation with other simulator components are essential.

ESRIG SYSTEM ARCHITECTURE

The ESRIG system is designed to execute concurrently the numerous software tasks needed for multi-mode radar simulation. For each mode simulation task, the ESRIG employs multi-stage pipeline-parallel processing. Within a pipeline stage there are one or more mode-specific execution threads , or sequence of function calls, which invoke the associated radar methods needed to process data. One or more software task stages are combined to form a process which is associated with a specific hardware module. Each process can be loaded into an individual processor since it a complete executable unit. A general purpose hardware environment with optimized data communication effectively supports this software architecture.

Hardware Environment

The hardware environment for the ESRIG is shown in Figure 1. Each box in the diagram represents a separate VMEbus processor. Each processor in the system contains a high-performance digital signal processor (DSP) chip, memory, and bus interface logic and is an autonomous unit. This hardware configuration allows flexibility in providing the proper data and computational flow needed to implement the ESRIG software architecture. It has been carefully designed to ensure proper computational, bus, and memory capacities to allow the simulation of a large range of radar systems through the use of additional processing elements and/or memory cards. It is also possible to extend the system into multiple backplanes if necessary.
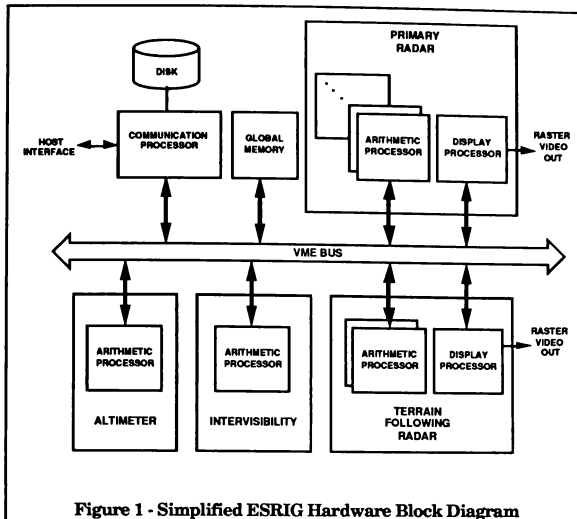
Figure 1 - Simplified ESRIG Hardware Block Diagram

Arithmetic Processor

The arithmetic processor (AP) is the heart of the ESRIG system. It is an Evans & Sutherland proprietary device based on a state-of-the-art DSP chip, and includes a large dynamic memory, fast static memory, and specialized VMEbus and VSBbus interface logic. Multiple APs can be included in the system to increase the total processing capacity available. A special memory access mode allows any processor on the VMEbus to broadcast information to all of the APs simultaneously in order to greatly reduce the amount of bus traffic and allow additional processors to be added with minimal system degradation.

Display Processor

The display processor (DP) is similar in design to the arithmetic processor; it is also a proprietary device based on the same DSP chip. It differs from the AP in that memory on the DP is configured as a dual frame buffer and video display circuitry is included. A separate video overlay memory is provided for display symbology.

The frame buffer provides continuous scan-converted output video on the DP, thus isolating the time base of radar image update from the time base of an external display synch master. Computational load variations or overloads (beyond normal sweep line time intervals) in range bin processing thus have no adverse effect on reliable display refresh and synch lock.

The DP is capable of driving a variety of displays using either standard or non-standard display formats. This capability is programmable and includes the ability to lock to an external video source such as a symbol generator or avionics display synch master. Supported video standards include EIA RS-170A and RS-343 for high-resolution displays. Direct video recording is also possible.

Communication Processor

The Communication Processor (CP) is an industry standard VMEbus single board computer. It includes Ethernet and serial external interfaces, a SCSI disk controller, and a large memory. The CP runs under the widely used VxWorks real-time operating system,

Figure 2 - ESRIG GMR Mode Functional Block Diagram
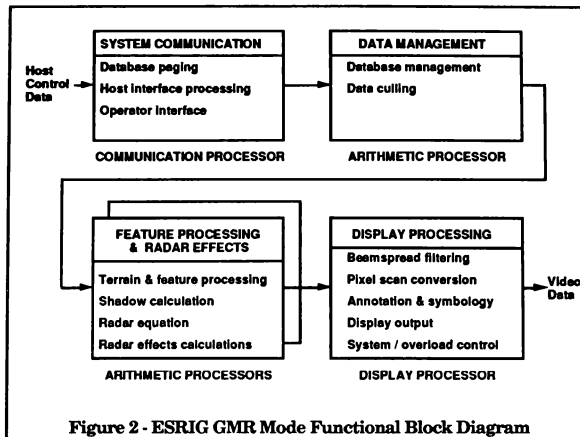
providing ready control of the SCSI disk, serial link, and external communications. The system also allows the addition of specialized interface cards which are readily available for VMEbus systems.

Global Memory

One or more VMEbus memory cards can also be added to the system to increase the amount of available on-line database storage. These cards contain both VMEbus and VSBbus interface logic to allow off-loading of VMEbus accesses to the VSBbus.

Software Architecture

The ESRIG system hardware has been specifically designed to support a scaleable pipeline-parallel software architecture. This architecture is extensible and is not tied to any single microprocessor or board implementation and so can take advantage of future hardware developments.

Figure 2 shows the basic partitioning of the radar simulation stages for the GMR mode task.

The CP provides system communication and utility functions as shown in the diagram. In particular it handles all external communications and database paging. The APs

perform database culling, as well as all feature and radar simulation processing except that which is beam-spread filtering and display related. The feature and radar effects processing are executed concurrently on multiple APs. Each of these arithmetic processors contains identical software and executes the same algorithms on different data.

The DP performs the display-related functions as shown in Figure 2. Filtering is started after a sufficient number of sweep lines have been received from the arithmetic processors. Each range bin output from the filtering process is scan converted into the frame buffer for presentation on a raster display or is sent directly to a vector display.

Software Design Philosophy

Because of the similarity among several radar modes, it is natural to design the real-time software in a modular fashion so that data and algorithms that are common to several modes can be reused. By designing and implementing the ESRIG software system in terms of objects (consisting of private data combined with procedures or methods operating on the data) a simpler more flexible system results. The object-oriented paradigm provides encapsulation of essential data, as well as dynamic binding of the desired method. This

facilitates incremental software design, coding, and rapid debugging.

Software Organization

Each ESRIG mode simulation is performed by a separate task. A task is itself broken into one or more sequentially executed pipeline stages. Each stage can be executed as a single process (possibly duplicated on several boards) or several stages can be combined into a single process and executed sequentially. For example, at least one GMR task stage is duplicated on many AP cards for parallel execution, while several TFR task stages are sequentially executed on a single TFR AP board. Thus, each stage is either a separate, complete process which can be loaded onto a hardware module or is linked with other stages to form a complete process. Each process resides on a single processor board and is a complete code executing one or more stages. For example, the GMR task consists of three processes which together are called a macrosector process. Most hardware modules execute a single process, but the CP will execute multiple processes using the real-time operating system. Control is provided for each task by a stage scheduler. Intra-process communication is by direct local memory addressing, while inter-process communication is performed using a system message handler.

A task stage consists of one or more stage threads, where a thread is chosen for execution depending upon ESRIG mode. For example, the radar equation stage will have individual GMR and target mode threads. Each thread consists of one or more object methods (procedures) used to manipulate the object data needed for the thread calculations. Objects are always compiled and linked into threads and stages.

The use of a task-stage-thread software hierarchy allows for efficient and simple control of the multi-mode requirements independently of the particular hardware configuration used.

RADAR DATABASE

A radar database has many similarities to, and a few significant differences from, a visual database. For example, the centimeter-wavelength microwaves employed in most airborne radars can produce beam effects quite different from light at optical wavelengths.

Also, because of the nature of time-resolved radar, range resolution is preserved over all range bins within the sweepline, while the azimuthal resolution decreases with range as the antenna beam diverges. This constant range resolution is distinct from the usual perspective shrinking seen in visual images. These and other differences can have profound effects on database correlation and management and require careful attention during radar database modeling and simulation.

In order to meet its functional requirements, the ESRIG stores and retrieves databases with distinct primitives: a gridded terrain elevation database, an analytic feature database, and a texture map database. These databases share some similarities to and several significant differences from those used for visual image generators. As with visual image generators, feature and terrain information in a given geographic area is generally covered in the database by several sets of data at distinct levels-of-detail (LOD). Each feature LOD contains geometric and texture data scaled to match the fixed range bin resolutions chosen at each GMR range scale. A uniform texel density at each range scale is provided. Such uniform densities of texture and features allow a constant average feature processing rate to be achieved.

The feature and texture databases differ from those used in visual simulation mainly in the types of primitives and their associated attributes. Visual image generator databases contain mainly collections of polygons and texture maps having color, shading, or other visually oriented attributes. The ESRIG feature database contains DMA-like analytic feature primitives. These are combined with texture maps which supply additional radar attributes. Terrain height grids are employed in a manner similar to modern visual image generators.

ESRIG Database Primitives

An analytic feature primitive consists of a data structure containing geometric data, such as position, orientation, radius, height, and feature footprint vector list; non-geometric data, such as radar attributes and special processing codes; and associated texture map information, such as texture map pointers and offset, orientation, and scaling data.

A texture map primitive consists of sampled high-resolution radar information such as surface material code, scattering area, height, directivity, and intensity modulation. Texture primitives provide a means by which each texel, in the limit, can represent a complete three-dimensional radar point feature, thus resulting in a very rich feature capacity.

There are four distinct geometric types of analytic features in the ESRIG design (see Figure 3): point, lineal, areal, and volumeal (volumetric). Each feature type requires a different geometric description. The description used for the first three surface types closely resembles that of DMA DFAD. Additionally, ESRIG analytic features contain many radar specific attributes and processing-specific data not found in standard DFAD data. The notion of volumeal features is specific to the ESRIG, and is employed to model three-dimensional volume features which are not ground-based, such as weather cells and chaff. Volumeal features share many similarities with ground analytic features, but they are volumetric in their geometric description and use voxel, rather than texel, maps to fill them.

In the ESRIG, features can be mixed independently to provide both generic and geo-specific geometry and texture. For example, a geo-specific city outline can be filled with generic "urban texture" containing reflectivity and height information.
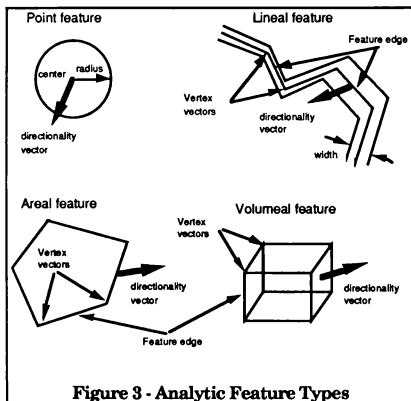
The ESRIG also provides the capability to use large unique global or repeating texture maps covering large geographic areas with either geo-specific, geo-typical, or generic texture maps for added database complexity in conjunction with or independently of an associated analytic feature.

The use of analytic features combined with texture detailing provides a feature-rich database in a very compact form. Radar simulation can be accomplished with either some type of non-textured analytic features or else purely gridded features. A database composed of non-textured purely analytic features requires more complex real-time processing if it is to display the same apparent feature density. A database using only analytic features suffers from a processing load level which varies from area to area as the density changes and therefore limits the complexity of the resulting image and possibly produces erratic behavior such as image breakup. A database using only gridded features, on the other hand, provides a rich feature density at a fixed processing rate throughout the database. It requires, however, a great deal of both disk and on-line storage, and the image can exhibit aliasing artifacts due to multiple sampling operations. Worse, important point features may be lost between samples.

The ESRIG combines these two approaches to provide the advantages of each and avoiding many of the disadvantages of either. Using analytic features reduces the storage requirements and, when combined with anti-aliasing techniques applied to avoid missing small point or lineal analytic features, ameliorates the sampling artifacts of a purely gridded approach. The addition of full radar attribute texture provides extremely high point feature density when required. Other advantages of this approach include ease in correlated modeling, the ability to modify a feature's LOD, and a close analogy to visual database primitives for better overall correlation.

Radar Correlation

A radar image generator is seldom used independently of other training components, such as a visual image generator and other sensor simulation equipment, and therefore must integrate into the overall system to provide

Figure 3 - Analytic Feature Types

a consistent tactical view to the subject. If this consistency, or correlation, is not achieved, the users will receive different cues than would occur in a real flight regime, and negative training will result. Ideally, all simulation subsystems would faithfully reflect the real world and this, of course, would provide exact correlation; however, this is not possible using existing technology. In general, both real world fidelity and one-to-one correspondence to the other simulation subsystems cannot be guaranteed. Correlation therefore needs to be defined as an appropriate compromise between achievable real-world fidelity and one-to-one correspondence to the other simulation subsystems, such as the out-the-window visual images. Correlation requirements vary for different training situations and even require the ability to use different strategies within the radar simulator system. This section describes the compromises which may be made and the reasons why the ESRIG provides the capability and flexibility to make these compromises and achieve correlation.

Sometimes a complete one-to-one correspondence between objects in the visual and radar databases is not always required for radar-important features. Differences in the simulation of the physics of radar energy reflection and visual light energy reflection lead to this somewhat paradoxical conclusion. For example, the size and amount of visual light reflected (visual cross-section) from a TV antenna may be insignificant at even relatively short distances. Even with specular reflection, the visual cross-section provides too little reflected light to matter. The antenna might thus be eliminated from a visual image generator's database, whereas its radar reflection (radar cross-section) as a dipole is still very strong at relatively large distances requiring that it be maintained in the radar database. Other examples of this effect occur with power line towers and open metal frame bridges whose visual areas are so small that they disappear from view due to perspective in the visual image. Since their radar cross-sections are large, however, they remain strong radar reflectors. Radar feature modeling and LOD transition ranges from high to low LODs therefore require different considerations than visual features. In this case, correlation is simply achieved by making the radar database a superset of objects contained in the visual

database and maintaining objects in the radar simulation to greater distances.

Making the radar database a superset of the visual database can also accommodate the differences in system capacities. The ESRIG is capable of achieving a high degree of realism through its use of analytic features and three-dimensional texture, while many visual image generators have a relatively low feature density (texture is used to boost this density). If the database is tuned and developed for the lower capacity system, the radar image generator capacity is not fully utilized. This may not achieve the desired image quality, especially when simulating Doppler-enhanced modes, nor provide an adequate training situation. The radar database should be augmented with additional features in order to maximize its training value and image quality. It is important in both these examples that even though one-to-one correspondence is not always needed, all important objects which are in both simulation systems are contained in the database and rendered as part of the image in the proper position. The development of the databases must be done in such a way that this is accomplished in a consistent manner at all LODs. Figures 4 and 5 depict correlated real-beam GMR and visual images.

The compromise between real-world fidelity and one-to-one correspondence is more difficult to make in determining the proper terrain resolution. If higher resolution terrain is used in the radar than that used by the visual system, the image will be more accurate, but important targets or features may be occulted differently between the two systems. In the GMR image this may not be as important a consideration as image fidelity, but in the intervisibility processing needed by the electronic warfare simulator it may be a dominant consideration. The pilot would not want to be hiding behind a ridge and to be shot down by a visually unseen enemy because the radar image generator was using a different terrain model. This may lead to subject confusion and negative training. One-to-one terrain correspondence is also critical in the case of terrain following radar where the pilot will be confused if the aircraft maneuvers around unseen or inconsistent terrain. The solutions here will vary depending on the requirements, but may include the use of different terrain databases for the TFR and intervisibility processing and the GMR

simulation, such that the TFR and intervisibility processing provides a one-to-one correspondence to the visual while the GMR produces a higher fidelity image. This requires a great deal of flexibility in the radar image generator.

If the radar image generator is inflexible in its ability to adapt to the correlation requirements, then correlation is costly, and difficult or impossible to achieve. The ESRIG provides the flexibility to span almost the complete range of correlation from real-world fidelity to one-to-one correspondence with the visual out-the-window image and other simulation subsystems. Its software-based approach allows it to easily adapt the algorithms used to vary the terrain resolution and feature density to match the correlation requirements. The system is capable of adapting different subsystems to matching real-world situations while other subsystems are maintaining one-to-one correspondence with the visual image generator.

Achieving correlation requires both a powerful database modeling system and a flexible radar image generator. A new modeling system developed for E&S image generator products, including the ESRIG, allows databases to be designed and built using a central master database. This central database contains a superset of all of the image generators' databases, and provides methods which allow the final database generation to contain all of the common objects plus those unique to a particular image generator. Correlation is made easier if the root database designs are similar. This is the case with the ESRIG database, in which the analytic object descriptions in the radar database can be easily derived from source data such as DMA data and are similar to those used in visual databases. It is also important that the database tools and formats be developed with a detailed understanding of both the visual and radar image generators. In this way the performance and training functions of both systems can be maximized.

CONCLUSION

The ESRIG provides a number of features that optimize its performance. It has been designed as a software-based system using an object-oriented style and executed on scaleable pipeline/parallel hardware. New database primitives have been developed to provide extremely rich feature densities. These primitives also support multi-sensor correlation and efficient modeling.

ACKNOWLEDGMENTS
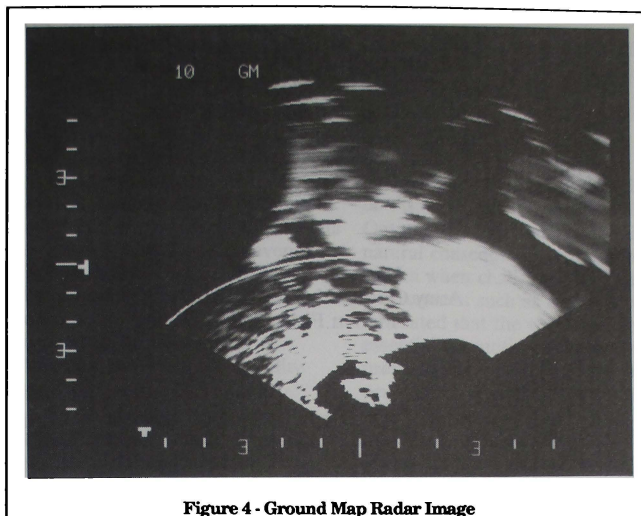
Figure 4 - Ground Map Radar Image
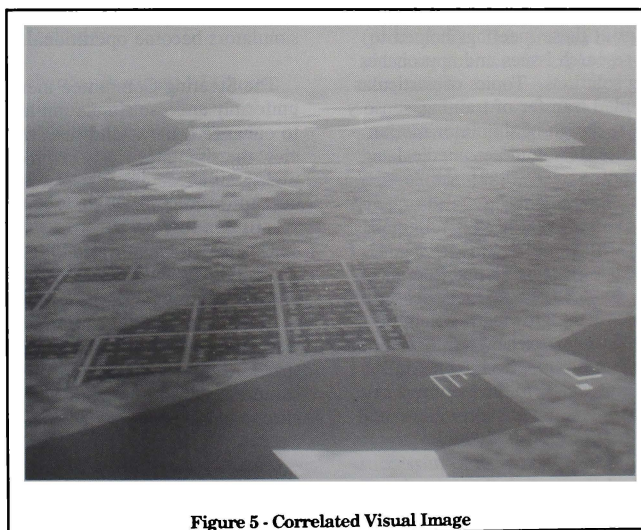
Figure 5 - Correlated Visual Image

RECENT PROCEEDINGS OF THE
NASA STEERING COMMITTEE ON
SIMULATOR INDUCED SICKNESS

Michael E. McCauley
Monterey Technologies, Inc.
Carmel, CA

Anthony M. Cook
NASA Ames Research Center
Moffett Field, CA

and

James W. Voorhees
Army Crew Station R&D Branch
Moffett Field, CA

Abstract

The Third and Fourth Meetings of the NASA Steering Committee on Simulator Induced Sickness were held in 1990 and 1991. The Steering Committee has 18 representatives from the U.S. Army, Navy, and Air Force as well as Canada and the U.K. The presentations and discussions at these meetings helped to focus on research issues and approaches to finding solutions. Topics of particular note included transfer of training, sensory conflict, the design of simulator motion bases, flight safety, and aeromedical management.

Background

The NASA Steering Committee on Simulator Induced Sickness was formed in 1987 to assess the growing problem of simulator sickness, exchange information on research programs, and promote efficient approaches to developing an better understanding of the problem and the potential solutions (McCauley and Cook, 1987).

In the first meeting, mounting international evidence was weighed by the Steering Committee and led to the consensus that simulator sickness is a problem for safety and training (see AGARD, 1988). Furthermore, the Committee members believe that the magnitude of the problem is likely to increase as more wide field-of-view simulators become operational.

The Steering Committee members uniformly endorse simulation as a method to enhance training and research. They view the simulator sickness problem as an undesirable side effect that we should attempt to minimize.

This paper summarizes the presentations and discussions of the Third and Fourth annual meetings of the Steering Committee. The summary of the discussions generally follows a topical structure. The information is greatly compressed and not necessarily chronological.

Guest Speakers

George H. Crampton

Dr. Crampton, Professor Emeritus at Wright State University, reviewed the highlights of 40 years of research on motion sickness (Crampton, 1990). Long ago, he found that visual motion can produce sickness in animals (cats) even when no actual motion occurs. This is seen as the same fundamental process leading to simulator sickness. Motion sickness is normal consequence of perceived motion, not some abnormality or aberration.

Dr. Crampton reviewed the neurophysiological processes involved in the development of motion sickness. He discussed the attractiveness of the Sensory Conflict Theory, but criticized the fact that it is not quantitative or predictive.

Greg L. Zacharias

Dr. Zacharias, President of Charles River Analytics, advanced a control systems modeling framework for understanding simulator sickness. Extending a traditional pilot-vehicle systems model allows one to substitute the simulator for the aircraft and to represent the pilot's perceptual and motor control loop processes. Residual cueing error (actual - expected) might represent the "conflict" basis for motion sickness. Adaptation, triggered when cue conflict exceeds some threshold, can be considered as a modification of the "expected" cues over time.

Dr. Zacharias advocates a control systems model as a way to support

quantification of the analysis of the simulator sickness problem and to help focus experiments.

Sheldon M. Ebenholtz

Dr. Ebenholtz, Distinguished Professor of Vision Science at SUNY College of Optometry, discussed asthenopia as a natural consequence of the human visual system when challenged by an unusual environment such as a simulator. He reiterated that the vestibular system is essential for motion sickness and, presumably, for simulator sickness. The oculomotor system serves to stabilize an object on the retina and works in concert with the vestibular system to distinguish between egocentric and exocentric sources of visual motion. The accommodation and vergence systems operate via feedback loops. Repeated need for correction of errors via feedback will result in modified feedforward signals (predictors). This process underlies the plasticity characteristic of perceptual adaptation.

Asthenopia is characterized by headache, eyestrain, and nausea (Ebenholtz, 1989). It is caused by inappropriate accommodation and slow adaptation. Illusions occur when the visual system modifies itself to compensate for a reflex system, such as the vestibular ocular reflex (VOR).

Dr. Ebenholtz suggested yet another definition of conflict, based on simultaneous reflex demands on the oculomotor system. The conflict usually can be resolved, for example, by successful visual tracking of an object. But these resolutions result in slip signals

(error) and can cause asthenopia (sickness). It is the price one pays for adapting to a new visual-intertial environment.

He reiterated the importance of proper calibration of the optical systems in simulators to ensure that the wavefronts are at optical infinity (or nearly so). He also suggested the possibility of strobing the visuals, to reduce vection and possibly, sickness.

Henry R. Jex

Mr. Jex, Chief Scientist at Systems Technology, Inc., summarized many years of research done by his firm in the areas of vehicle-control systems, motion effects, and kinetosis (the term he advocates for general motion sickness, including simulator sickness).

Studies on the effects of motion sickness on the performance of several tasks, including the "critical tracking task" indicate very little decrement until severe nausea and vomiting. Highly motivated subjects seem capable of literally "tracking while retching." Biodynamic interference also may cause performance decrement during motion. It is difficult to determine performance decrements in research settings because any motion-induced decrements are counter to learning effects.

Experience at STI with automobile driving simulators has indicated some problems with simulator sickness. The probability of motion discomfort from driving scenes appears to be strongly related to screen size. For example, no complaints are associated with normal home television (less than 20 deg FOV), occasional complaints are found with thee UCLA and the GM car simulators (40-60 deg FOV) and considerable discomfort is associated with Cinerama and Disneyland Circle-Vision (150-over 180 FOV). Display lags also seem to contribute to the problem of sickness in driving simulators.

Mr. Jex underscored the importance of a model to represent the processes involved in kinetosis. He has developed a comprehensive dynamic model for kinetosis (Jex, Riedel, and Smith, 1982). He suggests that motion sensing and postural control are highly evolved in pre-vehicular animals to enable locomotion through an inertially fixed 1-G world. Kinetosis may result from maladaptation of evolved animal locomotion subsystems to modrn moving-vehicle frames -- it is an artifact of civilization.

James R. Lackner

Professor Lackner, from the Ashton Graybiel Spatial Disorientation Laboratory at Brandeis University, discussed relevant research from his laboratory and reviewed sensory conflict theory.

Sensory conflict theory has several weaknesses. It does not relate to any known physiological process. Adaptive process are ongoing, based on resolving conflicts. Rarely does this involve sickness. The relationship between conflict and sickness is unclear. One definition of conflict is based on the difference between perceived input and a neural store. This concept seems strained

when one considers the nature of a neural store that would catalogue all prior inertial experience. Certain stimulus situations, such as the "BBQ spit" rotation, are quite successful at inducing sickness, but have no apparent conflict. It involves constant otolith stimulation and, at a constant rotational velocity, no stimulation of the semicircular canals. Again, the sensory conflict theory fails to provide insight.

Dr. Lackner reiterated that the vestibular system appears to be essential for motion sickness. Blind people get sick but labyrinthine defectives do not.

Flight simulator visual systems produce optokinetic stimulation and velocity storage (refs). The so-called "dumping" of velocity storage may be related to the development of simulator sickness (DiZio and Lackner, 1991).

In a simulator, the visual system "tricks" the body into thinking it is rotating, so compensation of self-induced coriolis forces should be occurring. This would be expected to result in the pilot trying to hold his head stationary to avoid coriolis cross-coupling (Guedry and Benson, 1978).

Changing the inertial mass of the head by wearing helmet-mounted displays can be expected to create changes in the motor control loops for head movement. As these changes create errors in intended head motion, sickness may result, at least until new motor control patterns are developed.

John O. Merritt

John O. Merritt of Interactive Technologies, Inc. has served for several years as a consultant to the Naval Ocean Systems Center, Hawaii Laboratory, for the development and evaluation of a teleoperated ground vehicle system for the Marine Corps. An operator, positioned in a stationary ground vehicle, controls a remote all-terrain vehicle. The operator wears a helmet-mounted display of stereo video imagery obtained from the remote vehicle. The pan and tilt of the remote cameras are controlled by the operator's head movement. Mr. Merritt reports that, indeed, simulator sickness (or perhaps "teleoperator sickness") does occur.

The occurrence of simulator sickness in a teleoperator system is to be expected. This configuration is similar to a pilot in a fixed-base flight simulator with a helmet-mounted display. The primary difference is that the teleoperator's imagery is from remote video cameras but the fight simulator imagery is computer generated.

Conflicting motion cues are inherent in the situation where an operator is sitting in a fixed (constant) inertial environment but seeing a dynamic wide FOV display that induces vection. In addition to this inherent visual-vestibular conflict, other factors may contribute to the observed sickness problem, such as servo lag and visual distortions (magnification).

Mr. Merritt discussed the advantages of stereo vision for interpreting terrain events, such as ditches and small cliffs.

He suggested that an earth-stabilized camera system mounted on the remote vehicle would improve operator performance and reduce the potential for sickness. He supported this contention with video tapes obtained in field trials. Mr. Merritt noted that head-controlled yaw of the remote cameras is influenced by tilt (vehicle roll attitude). Similarly, head-controlled pitch of the cameras is influenced by the pitch motion of the vehicle. He reported anecdotal evidence that a gravity-stabilized camera on the remote vehicle is less provocative of motion sickness in the teleoperator.

When the teleoperator is in a moving vehicle, the problems are magnified. The operator's visual and vestibular inputs are uncorrelated. His visual input dynamics relate to the remote vehicle but his vestibular input derives from the vehicle in which he is riding. The oculomotor control system is severely challenged by this configuration.

Research Updates

Army CSRDF

Thomas J. Sharkey of Monterey Technologies, Inc. presented an overview of the VISMOSYNC study done on the NASA Vertical Motion Simulator (VMS) (McCauley, Hettinger, Sharkey and Sinacori, 1990) and on the VISFLOW study done on the Army Crew Station R&D Facility simulator.

The four motion base conditions (including a no-motion control condition) in the VISMOSYNC study led to no statistically reliable difference in pilot ratings of simulator sickness symptoms.

The magnitude of individual differences in susceptibility made it difficult to detect these equipment manipulations with only 12 pilots per condition.

Mr. Sharkey reported that in the VISFLOW study, a significant effect of Global Visual Flow (GVF) was found, even though the study was shortened due to Operation Desert Shield. GVF was defined as the velocity divided by the altitude (eye height). For equivalent flight profiles, including velocity (ground speed) the GVF was varied by a factor of four by setting the altitude at 100 versus 400 feet. Significantly more simulator sickness was found at the lower altitude (higher GVF).

Dr. James C. Miller, consultant to Monterey Technologies for the Army CSRDB research program, presented the preliminary results of the physiological data collection and analysis. Six physiological signals were sampled at 100 Hz: electrocardiogram (ECG); forearm skin conductance level (SCL); electrogastrogram (EGG), chest circumference for ventilatory rate (f_v), fingertip pulse (BVP), and fingertip skin temperature (T_{sk}). In addition, the ECG signal was fed into a vagal tone monitor (VTM; Delta-Biometrics) which calculated in real time a component of respiratory sinus arrhythmia, as an index of parasympathetic activity. The VTM also provided a cardiotachometer function from which we extracted heart period (HP), the reciprocal of heart rate.

Group mean data suggest that higher motion discomfort ratings were associated with hypergastria (4-9 cpm EGG activity) and higher heart rates. Discriminant

analysis suggested that no single physiological variable dominates in the prediction of motion discomfort; physiological variables may predict motion discomfort when restricted to within-subject (idiosyncratic) comparisons; physiological variables may not predict motion discomfort when combined across subjects. These findings are not inconsistent with Cowings et.al. (1986). A full report of the physiological measurement program will be forthcoming.

Optical Flow and Edge Rate

Dr. Lawrence J. Hettinger reviewed a series of studies done previously at Ohio State in which the roles of Global Flow Rate and Edge Rate in pilot perception of altitude and velocity changes were evaluated (Owen, 1982). He hypothesized that vection (the illusion of self motion) is necessary to elicit simulator sickness. Previous studies have indicated that Flow Rate and spatial frequency are important factors in creating vection. Therefore, hypothesizes that flow rate and spatial frequency also may be critical for simulator sickness.

The magnitude of the vection illusion is proportional to optical velocity and edge rate up to about 120 deg/sec. There are large individual differences in the vection illusion. People who are insensitive to vection may be less susceptible to motion sickness. The requirement for high spatial frequencies in simulation is probably task dependent. That is, it may support high-resolution tasks such as target detection and recognition, but may not be necessary for aircraft control. In a

"smart" image generation system, spatial frequencies could be modulated to optimize task performance while minimizing the probability of sickness. Such advanced concepts will require further explication of the underlying relationships.

Canadian DCIEM

Dr. Lochlan E. Magee described a joint research program on simulator sickness conducted by DCIEM and the University of Toronto. The study focused on the variability of delays in the motion system and the visual system in a 747 aircraft simulator. Physiological and workload measures (NASA TLX) were obtained, in addition to symptom ratings and postural stability.

Experienced and novice pilots, all of whom were not experienced in the 747 aircraft, were used as subjects. Delays ranging from 90 to 330 msec. were introduced for the visual and the motion systems. Variability of the delays was introduced, ranging +/- 30 msec. from the nominal delay. Total flight exposure was about 90 minutes in two flights.

Results based on pilots self reports of simulator sickness indicated no significant effect of experience levels, groups (delay conditions), or flight condition/ sequence. Pilots workload ratings, however, were sensitive to delay. Longer visual delays led to higher ratings of workload. Although the independent variables had little apparent effect on simulator sickness, 30 of the 32 pilots reported some degree of "After-Sensations of Motion," and several reported delayed

onset (up to 24 hrs.) of headache and fatigue.

Multivariate Computational Methods

Glenn O. Allgood of the Oak Ridge National Laboratory presented an approach to predicting simulator sickness based on advanced computation methods. This approach was based on identification of a large number of potential predictor variables (45) including elements of an individualized whole body energy absorption model, motion spectrum data from the three linear axes of the simulator motion base, pilot flight hours, hours of sleep previous night, and so on. The data were submitted to a machine-learning method employing an inductive inferencing model which recursively calculated fits to the data, finding the sample attribute accounting for the largest information.

Mr. Allgood claimed that this advanced computational methodology was highly successful in classifying simulator sickness outcomes. The relationship of this process to predicting sickness in fixed-base simulators is unknown.

U.S. Navy Data Base

Dr. Robert S. Kennedy, Vice President of Essex Corp., reported on further analysis of the Navy data base from the survey of 10 simulator sites (Kennedy, et al, 1989). Factor analysis of the symptom ratings revealed three factors, tentatively named visual, vestibular, and vagal. Kennedy's hypothesis is that the scoring profiles provide information supporting the diagnosis of the source of the problem in specific simulators.

The example was given of two similar simulators with nearly equivalent sickness indices overall, but the constituent profiles differed.

Monitoring the symptom profiles over time can be accomplished by having pilots in training routinely complete the symptom questionnaire after each simulator session. This would enable the effectiveness of interventions to be assessed. Routine monitoring enables large data bases to be developed, which is necessary because the equipment effects are weak relative to other sources of variability, such as individual differences.

Discussion Topics

Human Visual Processing

Dr. Robert T. Hennessy, President of Monterey Technologies, Inc. and former student of H. Leibowitz, presented an overview of the distinction between "ambient" and "focal" visual channels. This notion, advanced by Leibowitz and colleagues at Penn State, posits two independent channels of visual processing serving different functions (Leibowitz and Post, 198?). The ambient system is fast, subconscious, and contributes to spatial orientation, postural equilibrium and movement detection. The focal system is slow, conscious, and supports pattern recognition and detailed analysis.

Dr. Hennessy suggests that the ambient visual system is important to understanding simulator sickness.

Dr. Edward Rinalducci, Professor of Psychology and Human Factors at the University of Central Florida, discussed

flicker, visual fatigue, and the perception of motion. The effect of flicker in causing visual fatigue can be measured physiologically even when subjects don't report perceiving flicker.

Recent trends in visual science make the distinction between two neural processing channels, called parvocellular and magnocellular. This functional classification is analogous to the ambient/focal distinction discussed by Dr. Hennessy. According to Dr. Rinalducci, Parvo = small and mediates the perception of high spatial frequencies and feature analysis. It is mediates color vision, sustained visual attention, and is relatively slow. Magno = large, and mediates movement perception, depth perception, and flicker. Magno is largely monochromatic, transient, phasic, and fast response. He suggests that further understanding of flicker and movement perception is needed in simulation display systems.

Motion Seats

Frank Cardullo, led a discussion of motion cues other than a motion base. Motion seats, often called "G seats," and similar devices are based on the premise that just a "jolt" may be sufficient to produce the sensation of motion, when accompanied by a dynamic visual display. Grant MacMillan has worked on G-seats at AAMRL and found that position and velocity combined (the SIGMA system) seemed to provide reasonable good motion cues. Motion seats are considerably less expensive than a full motion base system. Would motion seats reduce visual-vestibular conflict and

simulator sickness? This seems to be a viable question.

Motion Base Fundamentals

John Sinacori led a discussion of motion base technology. The basic premise of a motion base is that the commanded input from the pilot is transformed and (usually) reduced because they are constrained in physical space. One issue is how big to build them -- higher fidelity requires larger amplitude. The NASA VMS, for example is housed in its own seven story building. Such a facility goes beyond the resources that can realistically be dedicated to training simulators.

Accelerometers can be mounted on the simulator cab to determine the fidelity of the actual motion relative to the commanded motion. Although research simulators, such as the VMS, typically have accelerometers, most training simulators do not. This lack of instrumentation makes acceptance test procedures and routine maintenance difficult. Usually only rudimentary checks are done, such as the equations, the drive logic, compensation, transforms and single-axis noise and frequency response.

The instrumentation and procedures necessary to ensure good motion cuing should be include in training simulators. This would enable periodic checks of frequency response and washout to be conducted independently from the motion production system.

Motion Base Panel

At the 1991 meeting in Orlando, a panel of experts discussed on motion base design and performance. Steering Committee members John Sinacori and Frank Cardullo were joined on the panel by James Danos of NTSC, Bruce Baker, and Soren LaForce of NASA Ames Research Center. Their discussion included the issue of whether a motion base should filter energy at 0.2 Hz (the nauseogenic region) or should reproduce aircraft motion with as great a fidelity as possible. Mr. Danos noted that this frequency range is right in the middle of pilot control inputs. The consensus seemed to be that a motion base should not add any energy in the 0.2 Hz region (i.e., in washout algorithms). Also, it is important to distinguish between design criteria for controllability versus sickness. In some cases (like the 0.2 Hz region) the two sets of criteria may be in opposition. The final criterion, according to Mr. Danos, should be based on human perception rather than on hardware engineering.

A rule of thumb for motion perception is that 0.1 G is approximately the "indifference" threshold when a pilot is task loaded. This is well above absolute thresholds for linear motion perception (Benson, et al, 1986).

Medical Management

Medical management of simulator sickness is perceived as a critical issue by the military medical community. When should a flight surgeon medicate a pilot to prevent simulator sickness? Does medication hide the effects of a poorly designed or calibrated simulator? And when should an aviator be grounded due a severe or long lasting case of simulator sickness? Can side effects or state-dependent learning make medicated aviators less effective in subsequent aircraft flight? These are issues that concern flight surgeons assigned to aviation units where simulator sickness is endemic.

Research Methodology

Methodological issues were discussed in simulator sickness research. The behavioral scientist often looks for statistical significance to determine whether effects may have occurred by chance. Such procedures call for large sample sizes, ie. a large number of samples from each person or a large number of people. But flight surgeons and flight safety/ operations personnel are concerned if only one pilot has a bad experience (such as disorientation) resulting from simulator sickness. The scientist tends to report data in terms of percentages and probabilities; the flight surgeon wants to know the number of pilots who suffered each complaint. The scientist is trained to accept the null hypothesis (all is chance) until strong statistical evidence allows it to be rejected. Flight surgeons are instructed to ground pilots who use certain medications when the incidence of the side effect is extremely low.

Premature Guidelines

Mr. Ric Armstrong, representing the U.S. Army HEL/PMTRADE (in charge of Army simulator procurement), summarized a workshop on simulator

sickness that was sponsored by
PMTRADE in November 1989. An
interest in simulator sickness arose at
PMTRADE because of some problems in
the M-1 tank simulator prototype. The
PMTRADE two-day workshop culminated
in small working groups tasked with
generating preliminary guidelines for
simulator specifications and use.

Mr. Armstrong reviewed those
PMTRADE preliminary guidelines for the
NASA Steering Committee and stirred
considerable objection and controversy.
The members of the Steering Committee
objected in particular to the
"Recommended Long Term Solutions"
which included items on motion base
bandwidth, transport delay, visual-motion
asynchrony, flicker, and a 0.2 Hz filter.
The members felt that there is insufficient
evidence to support these items as
recommended solutions. For certain
items, such as a transport delay less than
50 msec, engineering oriented members
of the Steering Committee suggested that
such a design goal was impossible with
current technology and might not be
achievable within 10 years. The Steering
Committee strongly suggested that, while
simulator engineering guidelines are
needed, they must be supported by data.
Dr. Voorhees pointed out that cost of
simulators could increase enormously if
manufacturers attempted to build to such
specifications. The Committee agreed
with Mr. Beger's (NAVTRASYSCEN)
suggestion that any guidelines or
recommendations should be promulgated
only after thorough cost-benefit analysis.

Interim Solutions

If simulator sickness is acknowledged as
a problem, then what are the solutions?
Many quick fixes have been advocated for
motion sickness and simulator sickness,
such as ginger root (or ginger ale),
petroleum jelly in the naval, eating more
food, eating less food, wrist bands, knee
bands, meditation, and medication.

It is not fully understood why some
people will become sick and others will
not when exposed to the same dynamic
environment. But individual differences
are common in all measures of human
perception, performance, and physiology.

While further information is needed to
understand the factors that contribute to
simulator sickness, we can be fairly
confident of the following interim
"solutions":

o **Fly higher** (reduce Global
   Visual Flow)
o **Fly shorter** (symptoms develop
   over time)
o **Fly smoother** (reduce
   maneuvering -- no one gets sick
   flying straight and level)
o **Fly repeatedly** (nearly all pilots
   adapt with repeated simulator
   sessions)
o **Eliminate spatial and temporal
   distortions** in the simulator by
   design, maintenance, and
   calibration

Future Activities

Anthony M. Cook, Convener of the
NASA Steering Committee, led a
discussion of a draft status report and

statement of needs. Annual meetings of the Steering Committee are planned, although no comprehensive program addressing simulator sickness exists in the U.S. military services.

References

AGARD, (1988). Motion cues in flight simulation and simulator induced sickness. AGARD-CP-433. Neuilly Sur Seine, France: Advisory Group for Aerospace Research & Development.

Benson, A.J., Spencer, B.A, and Stott, J.R.R. (1986). Thresholds for the detection of the direction of whole-body, linear movement in the horizontal plane. Aviation, Space, and Environmental Medicine, 57, 1088-1096.

Cowings, P.S., Suter, S., Toscano, W.B., Kamiya, J., and Naifeh, K. (1986). General autonomic components of motion sickness, Psychophysiology, 23:542-551.

Crampton, G. (1990). Motion and Space Sickness. CRC Press, Boca Raton, FL.

DiZio, P. and Lackner, J.R. (1991). Motion sickness susceptibility in parabolic flight and velocity storage activity. Aviation, Space, and Environmental Medicine, 62, 300-307.

Ebenholtz, S.M (1988). Sources of asthenopia in Navy flight simulators. Report No. TCN-87-635. Orlando, FL: Naval Training Systems Center.

Guedry, F.E. and Benson, A.J. (1978). Coriolis cross-coupling effects: Disorienting and nauseogenic or not? Aviation, Space, and Environmental Medicine, 49, 29-35.

Jex, H.R., Riedel, S.A., and Smith, J.C. Development of a comprehensive system model for motion/space sickness. (1982). TR-1157. Hawthorne, CA: Systems Technology, Inc.

Kennedy, R.S., Lilienthal, M.G., Berbaum, K.S., Baltzley, D.R., and McCauley, M.E. (1989). Simulator sickness in U.S. Navy flight simulators. Aviation, Space & Environmental Medicine. 60:10-16.

Liebowitz, H. and Post. R.B. (1982). The two modes of processing concept and some implications. In J. Beck (Ed.) Organization and Representation in Perception. Erlbaum.

McCauley, M.E. and Cook, A.M. (1987). Simulator sickness research program at NASA Ames Research Center. Proceedings of the Human Factors Society 31st Annual Meeting. (pp. 502-504). Santa Monica, CA: Human Factors Society.

McCauley, M.E., Hettinger, L., Sharkey, T.J., & Sinacori, J.B. (1990, September). The effects of visual-motion asynchrony on simulator induced sickness. AIAA-90-3172. American Institute of Aeronautics and Astronautics, Flight Simulation Technologies Conference, Dayton, OH.

Owen, D.H. (1982). Optical flow and texture variables useful in simulating self motion. Interim Technical Report, 2/1/81 to 1/31/82, Air Force Office of Scientific Research, Bolling AFB, Washington, D.C. AD-A117-016.

WHAT NEEDS DOING ABOUT SIMULATOR SICKNESS?

Robert S. Kennedy
Essex Corporation
Orlando, Florida

Michael G. Lilienthal
Naval Air Systems Command
Washington, D.C.

Jennifer E. Fowlkes*
Essex Corporation
Orlando, Florida

Abstract

The usefulness of simulators may be compromised by a phenomenon known as simulator sickness. Until research determines how to design simulators that produce no or acceptably low incidence of sickness, there are at least two issues which require attention: (1) simulator monitoring techniques to identify when simulation systems begin to produce unacceptably high levels of sickness, and (2) identification of crewmembers who are at risk for simulator-induced posteffects. This paper describes the development and implementation of a free-standing device that utilizes human output (i.e., symptomatology) to address two questions: (1) is the simulator sick? and (2) is the crew-member sick? The first question is a systems engineering question and pertains to quality assurance testing of simulators. The incidence of simulator sickness symptomatology can be tracked over time for a given simulator using a "quality control" model to detect shifts in calibration or other gradually emerging problems. The second question pertains to bio-medical evaluation; crewmembers who exhibit extreme reactions during simulator training are at risk for posteffects and need to be identified so they can be warned with regard to post-training activities. A fielded prototype system has demonstrated that such a system can have: (1) sensitivity to factors which may be expected to affect systems performance, (2) economy in terms of cost and crewmember time, (3) high reliability, and (4) good user acceptance. The profile of the symptomatology holds promise for identifying and targeting the equipment features which, when fixed, will alleviate the problem. A recommendation is made that a technical data base be assembled from a series of field experiments where "naturally occurring" changes to the equipment be monitored "pre," "per," and "post" modification in very large samples (> 100) of pilot exposures.

Introduction

Flight simulators provide an essential role for training in both military and commercial environments. They are utilized to train crewmembers for almost every conceivable operational platform (both fixed and rotary wing aircraft, spacecraft, tanks, automobiles, etc.) (Haber, 1986), and their benefits in terms of cost, safety, and flexibility assure that they will remain an integral part of training. It has been estimated (Shelsby, 1989) that the simulation market, at about $2.49 billion in

1987, will be close to $6.23 billion in 1992. Accompanying the growth of simulation in training has been increasing sophistication in simulation technology. Beginning with the Link "blue box" in the 1930s and progressing to complex systems encompassing six-degree-of-freedom motion bases and detailed, near-photographic quality visual display systems. Flight simulators are also invaluable for evaluation and development of new systems in the safer, better controlled, and less expensive ground-based environment.

However, the usefulness of innovation in simulation technology may be compromised by a phenomenon known as simulator sickness. An attendant consequence of simulation, even from its earliest days (Miller & Goodson, 1958; Havron & Butler, 1957), has been the occurrence of motion sickness-like symptoms in pilots and crewmembers. These symptoms, which include nausea, stomach awareness, disorientation, eyestrain, and headache have been reported by persons training in all forms of vehicular simulation (automobile, aircraft, tank), but have been most extensively documented in personnel training in military flight simulators. The problem is likely to be aggravated with advances in all forms of virtual environment technology and with trends for representing simulated environments in a more realistic fashion. There is evidence that simulator sickness may negatively affect attitudes of crewmembers towards simulators, and that it may jeopardize crewmember safety and reduce readiness because of restrictions imposed after training.

A combination of factors are implicated in simulator sickness -- from simulator equipment features (e.g., type of visual display) to the state of crewmember fitness. With little doubt, simulator visual display systems, and the interaction of visual with motion cuing, play an important role in the production of the malady. However, engineering methods to reduce simulator sickness are still imperfect. Until research determines how to design simulators that produce no or acceptably low incidences of sickness, there are at least two issues which require attention: (1) simulator monitoring techniques to identify when simulation systems begin to produce unacceptably high levels of sickness, and (2) identification of individuals who are at risk for simulator-induced posteffects.

The objective of this paper is to describe the development of an empirical approach to the study of simulator sickness using a free-standing portable computer. A unique attribute of the device is that it utilizes human output (i.e., reports of symptomatology) to addresses

---

*Dr. Fowlkes is now located at Engineering and Economics Research, Inc., Orlando, Florida.

489

two primary questions: (1) is the simulator sick?, and (2) is the crewmember sick? The first question pertains to quality assurance (QA) testing of simulators. Based on crew-members' reactions (i.e., symptomatology) to a simulator over a period of time, levels exceeding a predetermined criterion level can be used to signal that a simulator has "gone out of bounds" and needs to be evaluated. The profile analysis of the symptoms can provide information about particular equipment problems. The second question pertains to biomedical management. Crewmembers who exhibit extreme reactions are at risk for simulator-induced posteffects and need to be identified so they can be warned or restricted with regard to post-training activities.

System Description

Simulator Sickness Questionnaire

The primary measurement tool implemented is the Simulator Sickness Questionnaire (SSQ) (Lane & Kennedy, 1988) which was developed with the following characteristics in mind: (1) sensitivity to simulator sickness, and (2) diagnostic capability such that the scale is informative about the separable dimensions of simulator sickness. As will be discussed, it is this latter characteristic which may be used to guide engineering fixes to simulators which produce unacceptably high levels of sickness.

The SSQ consists of a checklist of 16 symptoms, each of which is rated in terms of degree of severity (see Table 1). A diagnostic scoring procedure is then applied to the checklist, resulting in a Total Severity (TS) score reflecting overall discomfort and scores on three separate subscales. Scores on the Nausea (N) subscale are based on the report of symptoms which relate to gastrointestinal distress such as nausea, stomach awareness, salivation, and burping. Scores on the Visuomotor (V) subscale reflect the report of eyestrain-related symptoms such as eyestrain, difficulty focusing, blurred vision, and headache. Finally, scores on the Disorientation (D) subscale are related to vestibular disturbances such as dizziness and vertigo.

An impetus for the development of the SSQ was a requirement to obtain information regarding the separable dimensions of simulator sickness symptomatology. A related objective for the development of the SSQ was to utilize symptom profiles as an indicator of the simulator components that were contributing to symptomatology. It is known that different symptom profiles can result from intersimulator differences (Kennedy, Lilienthal, Berbaum, Baltzley, & McCauley, 1989). Different profiles might be indicative of which characteristics of the stimulation were producing the unwanted effects. Likewise, profile changes within a simulator may be indicative of an emerging problem and diagnostic of its cause (Hettinger, Lane, & Kennedy, 1988; Lane et al., 1988).

The breakout of the SSQ subscales, reflecting the separable dimensions of simulator sickness, is based on the results of a factor analysis of over 1000 cases of symptomatology reports by pilots training across 10 Navy and Marine Corps simulators representing a variety of configurations and platforms (Lane et al., 1988). These subscales or dimensions appear to operate through different "target" systems in the human to produce undesirable symptoms. Normative data are avilable so that the incidence in different simulators can be compared.

System Configuration

A description of the general system configuration is shown in Figure 1. The SSQ is implemented on a free-standing portable computer (Zenith 171). In practice, crewmembers respond to the SSQ after every simulator flight. They also enter other relevant background information which might include: role during the flight (i.e., pilot or copilot) and state of fitness. The total time requirement is generally less than 3 minutes.

Symptomatology is automatically scored according to the SSQ scoring procedures (Lane & Kennedy, 1988). Thus, immediate feedback is provided. If an individual's score exceeds a criterion value, a management decision can be made regarding whether post-training restrictions are necessary. Criterion values can be based on previously collected simulator sickness data or on operational or economic goals. Data, which are time and date stamped, are automatically saved and may be processed in a number of ways which are elaborated upon in the next section.

Applications

A variety of applications of the system include systems-engineering, monitoring of training regimes, biomedical management, and as the basis for an R&D data base.

Systems-Engineering

Simulator sickness occurs because humans are sensitive to sensory conflicts and other provocative aspects of simulation. Thus, for example, if a simulator runs out of tolerance on some factor (e.g., increased optical distortion, too much very-low frequency vibration in the motion spectrum) that increases sensory conflict, the measurement of symptomatology reported by individuals who use the system should reflect this change. Thus, human performance (i.e., rating of symptomatology) may be used to monitor systems performance. Figure 2 illustrates how responses (SSQ Total Severity score) may be shown over simulator runs to monitor how a trainer is functioning. It can be seen in this example that the SSQ Total Severity score remains stable for the first 100 or so runs and then exceeds a predetermined criterion level of sickness severity. The change of sickness level serves as a flag that an evaluation of the simulator is warranted. Evaluation may include the collection of engineering data and other human performance data. In addition to these efforts, as noted above, the SSQ Total Severity score can be decomposed into the three separable dimensions to examine symptom profiles.

TABLE 1. HARD COPY OF SIMULATOR SICKNESS QUESTIONNAIRE

Indicate below the severity of each symptom as it applies to you right now.

General Discomfort	None	Slight	Moderate	Severe
Fatigue	None	Slight	Moderate	Severe
Headache	None	Slight	Moderate	Severe
Eye Strain	None	Slight	Moderate	Severe
Difficulty Focusing	None	Slight	Moderate	Severe
Increased Salivation	None	Slight	Moderate	Severe
Sweating	None	Slight	Moderate	Severe
Nausea	None	Slight	Moderate	Severe
Difficulty Concentrating	None	Slight	Moderate	Severe
Head Fullness	None	Slight	Moderate	Severe
Blurred Vision	None	Slight	Moderate	Severe
Dizzy (Eyes Open)	None	Slight	Moderate	Severe
Dizzy (Eyes Closed)	None	Slight	Moderate	Severe
Vertigo	None	Slight	Moderate	Severe
Stomach Awareness	None	Slight	Moderate	Severe
Burping	None	Slight	Moderate	Severe

Training

Changes in simulator utilization may dramatically change the incidence of simulator sickness. Previous work has shown that modification of a training syllabus dramatically reduced the severity of simulator sickness (Fowlkes, Kennedy, Lilienthal, & Dunlap, 1989). Conversely, the addition of new syllabus objectives, changes in training schedules or the length of training sessions, etc., may result in increases in symptomatology that are unacceptably high. The device offers a method by which the effects of such changes on crewmembers may be monitored. Additionally, if the profile of side-effects from training does not match a profile which may occur with aircraft usage, this can serve to signal differences in the trainers vis a vis the aircraft.

Biomedical Management

The occurrence of simulator sickness is not restricted to the period of exposure. Symptoms may outlast the stimulus by several hours, and occasionally, days. Baltzley and colleagues (Baltzley, Kennedy, Berbaum, Lilienthal, & Gower, 1989) reported that at least one in five pilots has experienced a symptom after leaving the simulator building. Posteffects include dizziness, postural instability, and flashbacks, all of which may compromise safety (Baltzley, et al., 1989; Crowley, 1987; Fowlkes, Kennedy, & Lilienthal, 1987). Flashbacks, which include illusory sensations of climbing and turning, sensations of negative g, and perceived inversions of the visual field, are particularly bothersome because of their sudden, unexpected onset. Individuals who exhibit extreme reactions during training are at risk for simulator-induced posteffects and need to be identified so they can be warned or restricted with regard to post-training activities. The device used here offers a technique to monitor crewmember responses and to identify those who show extreme reactions to the simulator. Post-training precautions may be applied to these individuals, thus improving the safety of select individuals and preserving the operational readiness of the majority of crewmembers.

R&D

Finally, data collected with an intended device can be a rich source of information regarding optimum flight schedules, adaptation effects, effects of engineering changes, etc., and may be used to form an R&D data base.

Results From Field Trials

A prototype device was fielded in October 1988 at NAS Whiting Field, Milton, FL, for the TH-57C training devices. Two TH-57C Flight Instrument Trainers with visual displays are currently in use; device "4" introduced in October 1988, and device "2" introduced in January 1989. The primary use of the simulator sickness recording device at this location is for biomedical management. After over a year in operation, several conclusions are evident:

Results

Figure 3 contains the distribution of scores that were obtained for >3600 cases in the TH-57C simulator. It may be seen that half the population reported essentially no symptoms for their exposure and half reported various amounts of sickness from mild to severe. Subsequent analyses were performed using arithmetic means and, because of the extreme skewness, 75th percentile scores. The rationale for using the 75th percentile was that this is essentially the midpoint of the portion of the population that was adversely affected by the exposure.

Figure 4 shows a comparison of the two simulators for the three types of symptoms and total score. Note that both simulators have virtually identical patterns of simulator sickness symptomatology; incidence being greater for

DEVICE FOR BIOMEDICAL
EVALUATION AND SYSTEMS ENGINEERING
FOR SIMULATORS

o Implemented on Zenith 181 Portable Computer
o Three Minutes to Complete

1. BACKGROUND
   INFORMATION
   ENTERED.

   FIT?

   Ⓨ   N

2. 16 SSQ SYMPTOMS
   RATED FOR SEVERITY

   NAUSEA
   ↓
   NONE SEVERE

3. SYMPTOMATOLOGY
   AUTOMATICALLY
   SCORED.

   SEE YOUR
   INSTRUCTOR

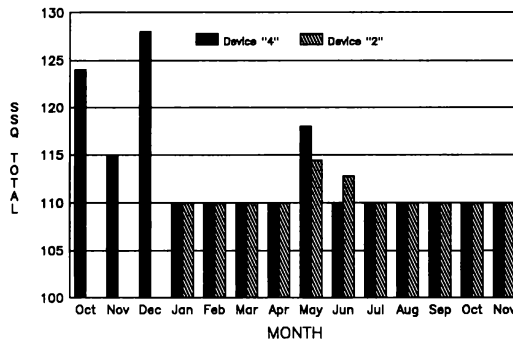Figure 1. General System Description of the Device.

75th Percentile SSQ Score

Figure 2. 75th Percentile SSQ Total Severity
Scores Depicted Over Time

things related to nausea and vomiting. Theory would imply that the chief problem in this simulator is likely to be connected with the motion environment rather than to either visuomotor or disorientation issues. One would expect that, if the visuals were particularly bothersome, there would be a greater incidence of visual epiphenomena. As this is not the case, solutions to simulator sickness in this device should attend to the motion base and perhaps the visual-inertial interactions.

Figure 5 shows the 75th percentile score for the period that the data were collected (from November 1989 until February 1991). It may be seen that after the simulators were installed, there was an initial settling-in period (reflected in relatively high symptomatology levels) followed by a relatively flat incidence level for the 75th percentile score over a 12-month period. In June and July of 1990, there was an increase in the sickness score. From discussions with personnel at the simulator site, there was a corresponding surge in simulator usage to the extent that Saturday flying and two hops per day were accomplished. The data from Figure 5 reflects this surge and, when the normal flight schedule was resumed, the incidence of sickness also went down. Because of the extreme stability of these measures, it is possible to use the score at the 75th percentile on a weekly basis in order to evaluate periods just before and just after maintenance to observe whether sickness rates rise before a maintenance period and/or fall subsequent to a maintenance period. These latter data are not yet available.

Figure 6 shows the incidence of sickness as a function of day of the week. It may be seen that day of the week has no influence on the incidence of simulator sickness. Figure 7 shows time-of-day effects. Sickness appears lowest after lunch, and perhaps suggests the importance of having food in the stomach. There is a slight increase in sickness at 1800 which may relate to missing the dinner hour. We might not expect this pattern of results in a simulator wherein the chief complaint is eyestrain.

Figure 8 reveals the effects of repeated hops on adaptation to simulator sickness stress. Note that on hop 1, the 75th percentile person exhibits a score of 115, which is comparable to the average score on the Navy's most nettlesome simulator. However, after four hops, most of the simulator sickness had subsided. Figure 9 shows the score obtained on each pilot's second exposure as a function of the separation between exposures. Note that sickness rates are highest when hops are on the same day or one day apart. Similarly, when hops are spaced more than five days apart, there is also little adaptation. The optimal spacing, in terms of controlling the incidence of simulator sickness, appears to be 2-5 days between hops.

Figure 10 shows a comparison of this simulator with others that have been examined by this scoring technique. It may be seen that this simulator has a higher incidences than 50% of the trainers previously examined.

Discussion

The present study reports findings from a semi-automatic reporting system whereby >3600 exposures have been recorded and analyzed. A new, normalized standard scoring method was applied to the symptomatology reports. This new procedure classifies symptoms according to three factors which were statistically derived from a large data base, but which also have a theoretical relevance since they cluster in accordance with somewhat disparate neural pathways. There are symptoms which are of "Neurovegetative," "Vestibular," and "Oculomotor" origin. There is also a "General Discomfort" or Total Score. Using this scoring method, we found that adaptation occurs over hops, so that after four hops sickness is very slight. The best regime for promoting this adaptation is to space flights two to five days apart. We noted that the two simulators appear to be essentially equal in their symptom profile, and this implied that the motion platform may contain some anomalies. It was found that day of the week was essentially without effect, but time of day had a small and perhaps intriguing effect. One of the interesting findings is that the 75th percentile metric can be a very stable index of activity in simulators and perhaps has some utility for monitoring maintenance of equipment during in-service engineering. We would also recommend that baseline scores should be obtained and then compared to the incidences and symptom mixture after any engineering changes are made to simulator configurations.

Because of the high between-subject variables and changing character of the response due to adaptation and other time-course factors, large samples are required for simulator sickness research. Since the field is the only place where such sample sizes exist, the authors proposed that field experiments be conducted to derive a technical data base. In this work, "experiments" would capitalize on the modifications to a simulator and record before and after change. The change could be proscribed syllabus differences (e.g., proportion of time at certain altitudes, hop length) as equipment updates (more resolution, higher refresh rates, shorter asynchronies) or naturally occurring maintenance-related "downs" (e.g., no motion, chin window out, etc.). By this method, a large data base can be assembled at low cost which can be interrogated posthoc to provide information for future design and upgrades. For example, if there is no difference in simulator sickness incidence whether motion is on or off and the statistical power of that analysis is very good (viz. > .9), one might suggest that decisions about motion/no motion should be made on other grounds. In one study, using this methodology, that is exactly the outcome which occurred (Kennedy & Smith, 1990).

References

Alexander, S. J., Cotzin, M., Hill, C. J., Jr., Ricciuti, E. A., & Wendt, G. R. (1945). Wesleyan University studies of motion sickness: V. Incidence of sickness at various hours of the day. Journal of Psychology, 20, 19-24.
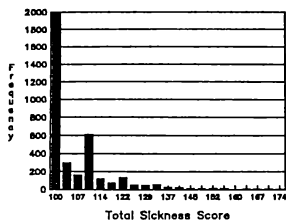
Fig. 3. Frequency Distribution of
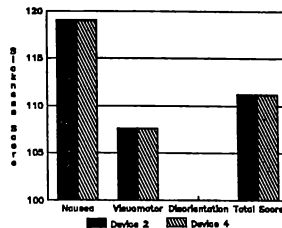Total Sickness Scores (N=3691)

Fig. 4. Symptom Comparison Between
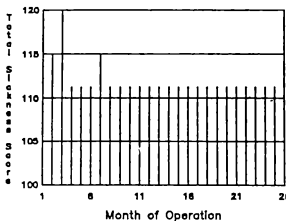Duplicate TH-57 Trainers

Fig. 5. 75th Percentile Sickness
Scores Across 26 Months
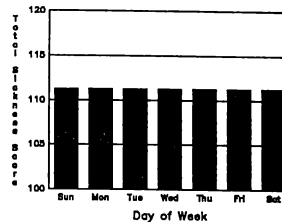of Data Collection

Fig. 6. 75th Percentile Sickness
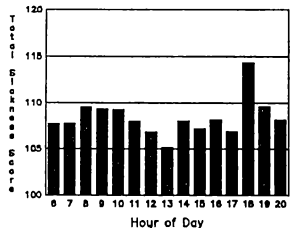Scores As a Function of
Day of Week

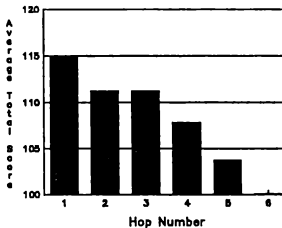Fig. 7. 75th Percentile Sickness Scores
As a Function of Time of Day

Fig. 8. 75th Percentile Sickness
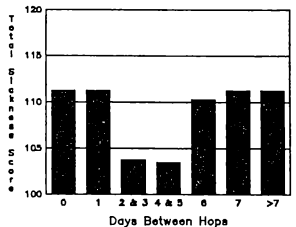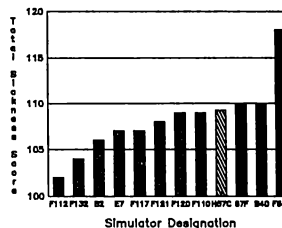Scores As a Function of
Hop Number

Fig. 9. 75th Percentile Sickness Scores
As a Function of Days
Between Hops

Fig.10. Simulator Sickness Compari-
son of 12 Navy/Marine Corps
Flight Trainers (Fixed-Wing
and Rotary-Wing)

Baltzley, D. R., Kennedy, R. S., Berbaum, K. S., Lilienthal, M. G., & Gower, D. W. (1989). The time course of postflight simulator sickness symptoms. *Aviation, Space, and Environmental Medicine*, 60, 1043-1048.

Bittner, A. C., Jr., & Guignard, J. C. (1988). Shipboard evaluation of motion sickness incidence. In F. Aghazadeh (Ed.), *Trends in ergonomics/human factors V*. New York: North Holland.

Crampton, G. (Ed.) (1990). *Motion and space sickness*. CRC Press, Boca Raton, FL.

Cronbach, L. J. (1990). *Essentials of psychological testing* (5th ed.). New York: Harper & Row.

Crowley, J. S. (1987). Simulator sickness: A problem for Army aviation. *Aviation, Space, and Environmental Medicine*, 58, 255-357.

Fowlkes, J. E., Kennedy, R. S., & Lilienthal, M. G. (1987). Postural disequilibrium following training flights. *Proceedings of the 31st Annual Human Factors Society* (p. 488). Santa Monica, CA: Human Factors Society.

Fowlkes, J. E., Kennedy, R. S., Lilienthal, M. G., & Dunlap, W. P. (1989). *Control of simulator sickness by simulator usage, adaptation, and other means*. Paper presented at the 60th Annual Scientific Meeting of the Aerospace Medical Association, Washington, D. C., May 7-11.

Gower, D. W., Lilienthal, M. G., Kennedy, R. S., & Fowlkes, J. E. (1987). Simulator sickness in U.S. Army and Navy fixed- and rotary-wing flight simulators. *Proceedings of the AGARD Medical Panel Symposium on Motion Cues in Flight Simulation and Simulator Induced Sickness*, Brussels, Belgium.

Haber, R. N. (1986). Flight simulation. *Scientific American*, 255, 96-103.

Havron, M. D. & Butler, L. F. (1957). *Evaluation of training effectiveness of the 2FH2 helicopter flight trainer research tool* (NAVTRADEVCEN 1915-00-1). Port Washington, NY: Naval Training Device Center.

Hettinger, L. J., Lane, N. E., & Kennedy, R. S. (1988). *Diagnostic measurement approaches to the problem of sickness in flight simulators*. Paper presented at the Meeting of the American Institute of Aeronautics and Astronautics, Atlanta, GA, September 7-9.

Kennedy, R. S., & Fowlkes, J. E. (1990). Simulator sickness is polygenic and polysymptomatic: Implications for research. *Proceedings of the IMAGE V Conference* (pp. 45-59). Tempe, AZ: IMAGE Society.

Kennedy, R. S., Lilienthal, M. G., Berbaum, K. S., Baltzley, D. R., & McCauley, M. E. (1989). Simulator sickness in U.S. Navy flight simulators. *Aviation, Space, and Environmental Medicine*, 60, 10-16.

Kennedy, R. S., & Smith, M. G. (1990). *Simulator sickness in two Marine Corps helicopter trainers: Influence of configuration, usage, and pilot history* (Final Rep., Contract No. DAAL03-86-D-0001). Research Triangle Park, Inc., Battelle, Inc.

Kennedy, R. S., Tolhurst, G. C., & Graybiel, A. (1965). *The effects of visual deprivation on adaptation to a rotating environment* (NSAM 918). Pensacola, FL: Naval School of Aerospace Medicine.

Lane, N. E., & Kennedy, R. S. (1988). *A new method for quantifying simulator sickness: Development and application of the simulator sickness questionnaire* (SSQ) (Tech. Rep. EOTR-88-7). Orlando, FL: Essex Corporation.

Miller, J. W., & Goodson, J. E. (1958). *A note concerning "motion sickness" in the 2-FH-2 hover trainer* (Research project NM 1701 11, Subtask 3, Report 1). Pensacola, FL: U.S. Naval School of Aviation Medicine.

Shelsby, T. (1989). Military turning to simulators to cut training costs. *The Sun*, August 13.

Ungs, T. J. (1987). *Simulator induced syndrome in Coast Guard aviators*. Unpublished Master's thesis, Wright State University, Dayton, OH.

THE EFFECT OF GLOBAL VISUAL FLOW
ON SIMULATOR SICKNESS

Thomas J. Sharkey and Michael E. McCauley
Monterey Technologies, Inc.
Carmel, California

ABSTRACT

An abbreviated experiment on the factors contributing to simulator sickness was conducted in the Army Crew Station Research and Development Facility at NASA Ames Research Center. Nineteen Army rotorcraft pilots from Fort Ord, California, participated in the study. Maneuvering Level and Global Visual Flow were manipulated in 40 minute scenarios. The pilot's task was to follow a lead aircraft through a long series of moderately aggressive turns in a river valley. The study was terminated early due to Operation Desert Shield. Despite the truncated sample size, Global Visual Flow had a significant effect on the incidence of simulator sickness, as determined by the pilot's symptomatology ratings. Flying the same profiles at lower altitudes (higher Global Visual Flow) resulted in greater sickness. These results have implications for managing simulator sickness through syllabus/scenario control.

INTRODUCTION

Simulator sickness occurs in many military flight trainers and research simulators (Kennedy, et. al., 1984; Kennedy, et. al., 1989). Simulator sickness is the manifestation of motion sickness- like symptoms (e.g., sea- or car-sickness) resulting from performing maneuvers in a simulator. It is assumed that flying the same maneuvers in an aircraft would not have resulted in the development of symptoms.

This syndrome is undesirable for a number of reasons including potential reduction in positive transfer of training between simulator and aircraft, and decreased user or institutional acceptance of research and training simulators. It is also possible that pilots employ strategies in a simulator to avoid or limit simulator sickness (such as minimizing head movements) which transfer from the simulator to the aircraft. The objective of our research program has been to understand the causes of simulator sickness so that flight simulators can be improved and used more effectively.

One of the most appealing potential explanations of simulator sickness is known as the conflict hypothesis (Reason & Brand, 1975). The conflict hypothesis predicts that, in a fixed base simulator, increased Maneuvering Intensity (MI) will lead to increased sickness. This prediction is based on the assumption that the amount of conflict between the visual and vestibular inputs would be larger during high intensity maneuvering than during more benign maneuvering.

The effect of changes which do not effect the magnitude of the disparity between motion signaled by the visual and the vestibular systems is not predictable from the conflict hypothesis. However, there have been suggestions that changes in the visual scene that do not alter the difference in motion signals are related to simulator sickness. In particular, the rate of Global Visual Flow (GVF) has been suggested as a factor in the perception of self-motion (Warren, Owen & Hettinger, 1982).

This experiment tests the prediction of the conflict hypothesis that increased MI will result in increased simulator sickness. It also examines the effect of changes in GVF on simulator sickness.

METHOD

Pilots. Sixteen volunteer U.S. Army helicopter pilots participated in this experiment. All of these men were based at Fort Ord, CA, and were assigned to operational units. Three of the pilots flew the simulator twice, once each at the low and high levels of GVF. Both flights of these three pilots was at the same level of MI. The other 13 pilots flew in only one GVF-MI combination.

Motion Sickness Susceptibility. The susceptibility of nine of the pilots to motion sickness had been estimated prior to their flights. The pilots rode in the rear of a mini-van while the van performed a series of S-turns. The windows of the van were not covered and the pilots sat facing forwards. The slalom was conducted at a speed of about 17 mph through cones spaced 200 ft apart. The van moved 12 ft to each side of an imaginary line between the cones during each cycle. After each pass through the cones, the van made a U-turn and went through the cones in the opposite direction. The slalom lasted a maximum of 15 minutes.

The pilots reported their well being on a 7-point scale before beginning the slalom, at 5 minute intervals during the slalom, and after the completion of the slalom. On this scale a "1" indicated that the pilot felt no symptoms, a "4" indicated moderate nausea, and a "7" indicated that the pilot was unable to continue due to discomfort. Intermediate values were used to report states of well being between these points.

The screening was done 1 to 4 weeks prior to the simulated flights.

Flight Simulator. The experiment was conducted in the U.S. Army's Crew Station Research and Development Facility (CSRDF). The CSRDF is located at the Ames Research Center, Moffett Field, CA. The CSRDF is a fixed base, advanced rotorcraft flight simulator (Lypaczewski et. al., 1986). It features a helmet mounted display (HMD) system with a wide instantaneous field of view and an unlimited field of regard. Figure 1 contains a schematic diagram of the HMD. The out-of-cockpit visual scene is generated by a General Electric Compuscene IV (Henderson, 1989).

The flight symbology used in this experiment is shown in Figure 2. The distance from the ownship to the lead aircraft (described below) was displayed in meters. Heading, altitude and velocity were displayed in degrees, feet, and nautical miles per hour, respectively.

Task. The pilots were briefed as to the purpose of the experiment, and their tasks described to them. Following the briefing, each pilot was fitted with a physiological sensor system. The preflight questionnaires and tests of postural equilibrium were completed prior to entering the simulator.
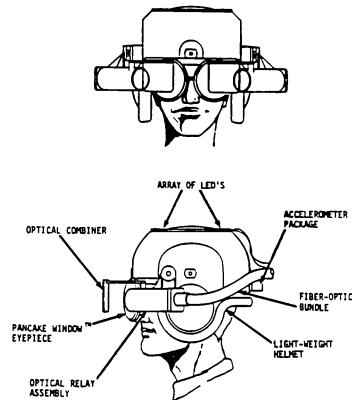

Figure 1.
Schematic diagram of the Helmet Mounted Display (HMD) system.

Immediately prior to the data collection period each pilot was allowed 10 to 15 minutes of "free flight" to familiarize himself with the handling and displays of the CSRDF. During this period the pilot was instructed to fly at 250 ft AGL until he felt capable of handling the aircraft. The pilot was then instructed to ascend or descend to the altitude he would be flying at during the experiment. Once at the proper altitude, the pilot was allowed to fly to the location of the lead aircraft. At the end of the training session the pilot positioned his own aircraft in a hover 250 to 350 meters directly behind the lead aircraft. Once in position the experimenter began the data collection portion of the flight.
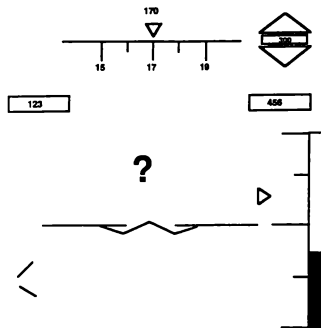
Figure 2.
Flight symbology presented
in the HMD.

During the data collection phase the pilots flew a loose trail formation behind a lead aircraft. The lead aircraft, a Soviet Hind helicopter, flew along a prerecorded route. The pilot was instructed to fly his aircraft along the ground track flown by the lead aircraft. Pilots were also instructed to remain at the same altitude as the lead aircraft and to remain between 250 and 350 meters behind. A digital display of interaircraft range (i.e., slant range) in meters was displayed to the pilot continuously in the Helmet Mounted Display. Whenever the range between aircraft was greater than 350 meters or less than 250 meters, an additional prompt to bring the range within tolerance was presented in the flight symbology.

The path of the lead aircraft was constructed by recording a flight of the CSRDF for later playback. The recorded flight was made at a nominal altitude of 100 ft AGL, and 100 kts airspeed. Two paths were recorded: (1) low maneuvering intensity (low roll rates and maximum bank angles) and (2) high maneuvering intensity. Higher altitude paths, which were to be flown at 400 ft AGL, were created by adding 300 ft altitude offsets to the original paths of the Hind.

The data collection phase of the flight required about 40 minutes.

Post-flight questionnaires and tests of equilibrium were completed after the flight and again approximately 30 minutes later.

Design. This experiment was designed as a 2 x 2 mixed design. The between-subject factor was Global Visual Flow (GVF) and the within-subject factor was Maneuvering Intensity (MI). Events in the middle east (Operation Desert Shield) limited the availability of pilots during the data collection phase of the experiment. This made it impossible to complete the experiment as designed. Consequently, the data was analyzed as a 2 x 2 randomized design. In order to facilitate analyses, the data collected on the first flight from each of the three pilots who flew twice was considered to be independent of the data collected on their second flight. That is, we treated the data from the second flight as if it was collected from a different pilot rather than as a repeated measure.

INDEPENDENT VARIABLES

Global Visual Flow. Global Visual Flow (GVF) has been defined as the velocity divided by the altitude where velocity is scaled in terms of altitude above the ground (Warren, Owen, & Hettinger, 1982). In this experiment GVF was manipulated by varying the altitude of the lead aircraft without changing its' ground track. In the high GVF condition pilots flew at approximately 100 ft AGL. In the low GVF condition the flight was at 400 ft AGL.

Maneuvering Intensity. Maneuvering Intensity (MI) was manipulated by changing the path of the lead aircraft. Frequency of direction changes, roll rates and maximum roll angles were larger in the high MI condition than in the low MI condition.

DEPENDENT MEASURES

Physiological Measures. The following physiological measures were recorded at 100 Hz:

Electrocardiogram
Vagal Tone
Electrogastrogram
Skin Conductance Level

Fingertip Skin Temperature
Fingertip Pulse
Ventilatory Rate

The technique used to collect and record this data, the results obtained, and interpretation of the results are discussed in Miller, et. al. (in preparation).

Self Reports of Well-being. Pilot's were prompted to indicate their well-being on a 7-point scale at 5 minute intervals. On this scale a report of "1" indicated that he was experiencing no symptoms, and a "7" indicated that he was unable to continue due to discomfort. Intermediate values were used to indicate intermediate states of well-being. Data from 12 pilots was available for these analyses. The effects of GVF and MI on the self reports of well-being were examined independently.

Simulator sickness Questionnaire. Each pilot completed a Simulator Sickness Questionnaire (SSQ) before flying the simulator, immediately after flying the simulator, and again 30 minutes after exiting the simulator. These questionnaires were scored using the A-technique proposed by Lane and Kennedy (1988).

Stand on Leg Eyes Closed (SOLEC). Pilots performed the SOLEC test 3 times before, immediately after, and again 30 minutes after their flight. The SOLEC task is described in Thomley, Kennedy, & Bittner, A.C., (1986). Pilots in this experiment performed the SOLEC standing on the leg the would stand on when kicking a ball. The maximum score is 30 seconds on each trial, and each pilot's score is the average number of seconds on three attempts.

Walk on Floor Eyes Closed (WOFEC). Pilots performed the WOFEC test three times before, immediately after, and again 30 minutes after their flight. Each pilot's score was the average number of steps on the three trials. This task is a slightly modified version of the "Walk-Heel-to-Toe" task described by Thomley et. al., (1986). The difference between our task and that of Thomley et. al. is that our procedure increased the maximum score from 10 to 12 steps on each trial.

Ballistic Pointing Task. Pilots performed a ballistic pointing task pre-, immediately post-, and 30 minutes post-flight. This task required pilots to extend a marker from the tip of their nose to a vertical target (i.e., a bull's eye) with their eyes closed. The target was located at nose height slightly less than an arm length in front of them. This task was performed three times at each administration, and the score was the average error of the three performances.

Three error scores were derived from the data and examined; radial, lateral, and vertical.

RESULTS

Self Reports of Well-being. Pilots flying in the high GVF (low altitude) condition reported worse symptoms than did the pilots in the low GVF condition. The mean in the high GVF condition is 5.00 compared to 3.14 in the low GVF condition. This difference is in the expected direction and is statistically significant (t_{12} = 2.10, p < 0.05, 1 tailed).

The difference in mean well-being reports of pilots flying in the high and low MI conditions are 4.11 and 4.00, respectively. This difference is not statistically significant (t_{12} < 1.0).

A two factor ANOVA was performed to investigate the possibility of an interaction between GVF and MI. The mean self reports are shown in Table 1. The interaction is not statistically significant ($F_{1,10}$ < 1.0).

TABLE 1
Maximum Self Reports
of Well-Being
MEAN (n)

		GVF	
		Low	High
MI	Low	3.667 (3)	4.500 (2)
	High	2.750 (4)	5.200 (5)

499

Simulator Sickness Questionnaire. The correlation between the total component of the post flight SSQ score ($SSQ_{(total)}$) and the maximum self report of well-being was 0.758 (p < .002, df = 14). The mean postflight $SSQ_{(total)}$ scores are shown in Table 2. Inspection of Table 2 indicates that the symptoms were most severe in the high GVF, high MI condition. The differences between the other three conditions were quite small. Table 3 summarizes a two factor (GVF x MI) ANOVA performed on this data.

Table 4 contains a summary of a 3 factor ANOVA performed on the $SSQ_{(total)}$ scores. The factor are GVF, MI, and Time of Administration (TIME). TIME (pre-, immediately post-, and 30 minutes post-flight) was the statistically significant factor. The mean $SSQ_{(total)}$ scores for pre-, immediately post- and 30 minutes post-flight are 105.6, 151.4, and 117.5, respectively.

TABLE 2
Mean Postflight $SSQ_{(total)}$ Score
MEAN(n)

```
                   GVF
         |--------|-------|
         |  Low   | High  |
|--------|--------|-------|
|  Low   | 144.9  | 143.6 |
|        |  (4)   |  (3)  |
MI|------|--------|-------|
|  High  | 132.7  | 176.3 |
|        |  (4)   |  (5)  |
|--------|--------|-------|
```
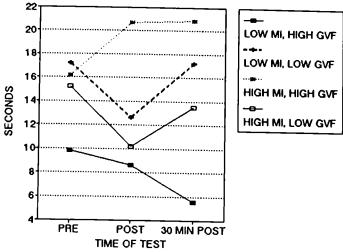

Figure 3.
Mean Duration on SOLEC Test.

A three factor (GVF x MI x TIME) ANOVA was performed on the data. The Anova is summarized in Table 5.

Walk on Floor Eyes Closed (WOFEC). Pilot performance on the WOFEC test is shown in Figure 4. Pilots in the high MI-high GVF group performed better post-flight than did pilots in the other conditions. A three factor (GVF x MI x TIME) ANOVA was performed on the data. The Anova is summarized in Table 6.

Ballistic Pointing Task. Separate three factor (GVF x MI x TIME) ANOVAs were performed on each of these measures. Figure 5, 6 and 7 show the average radial, vertical and horizontal pointing errors, respectively. Table 7, 8 and 9 summarize the corresponding ANOVAs. Positive values indicate that the error was above the target in the vertical error

Table 3
ANOVA Summary
Postflight $SSQ_{(total)}$ Scores

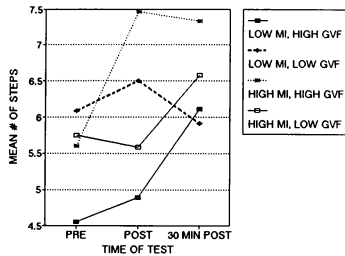| Source	SS	df	MS	F	p
GVF	1728.992	1	1728.992	1.283	< 0.30
MI	406.754	1	406.754	0.032	N.S.
GVF x MI	1938.085	1	1938.085	1.438	< 0.30
ERROR	16170.215	12	1347.518	-	-

Stand on Leg Eyes Closed (SOLEC). Figure 3 shows the mean performance on the SOLEC before, after, and 30 minutes after the simulator flight for each combination of MI and GVF. Only the high MI-high GVF group performed better post flight than they did preflight.

scores, and indicate that the error was to the right of the target in the horizontal error scores.

Maximum Self Report of Well Being and Screening Test. Self reports of well-being were available from nine pilots from both the van slalom and the simulated flights.

500

TABLE 4
SSQ$_{(total)}$ ANOVA SUMMARY

Source	SS	df	MS	F	p
GVF	1241.2500	1	1241.2500	1.188	< 0.30
MI	333.3266	1	333.3266	0.319	N.S.
GVF x MI	1239.9799	1	1239.9799	1.186	< 0.30
ERROR(w)	1926.0508	12	160.5042	-	-
TIME	16250.8655	2	8125.3428	16.859	< 0.001
GVF x TIME	809.1533	2	404.5766	0.839	N.S.
MI x TIME	140.2016	2	70.1008	0.145	N.S.
GVF x MI x TIME	847.8024	2	423.9012	0.880	N.S.
ERROR(b)	621.7814	24	25.9076	-	-

Figure 4
Mean Number of Steps
on the WOFEC Test
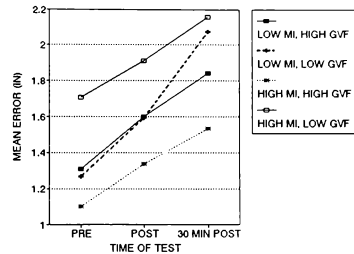
Figure 5
Average Radial Error in the
Ballistic Pointing Task.

TABLE 5
SOLEC ANOVA SUMMARY

Source	SS	df	MS	F	p
GVF	5.6694	1	5.6694	0.035	N.S.
MI	211.7637	1	211.7637	1.319	< 0.30
GVF x MI	560.5523	1	560.5523	3.492	< 0.09
ERROR(w)	1926.0508	12	160.5042	-	-
TIME	21.7534	2	10.8767	0.420	N.S.
GVF x TIME	91.1360	2	45.5680	1.759	< 0.20
MI x TIME	26.7225	2	13.3613	0.516	N.S.
GVF x MI x TIME	54.8919	2	27.4460	1.059	< 0.37
ERROR(b)	621.7814	24	25.9076	-	-
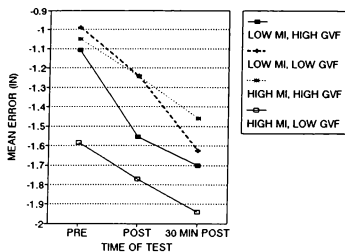
Figure 6.
Average Vertical Error in the
Ballistic Pointing Task.
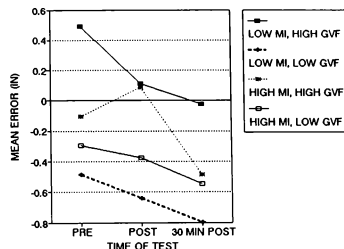
Figure 7.
Average Horizontal Error in the
Ballistic Pointing Task.

TABLE 6
WOFEC ANOVA SUMMARY

Source	SS	df	MS	F	p
GVF	0.0686	1	0.0686	0.007	N.S.
MI	5.8578	1	5.8578	0.589	N.S.
GVF x MI	9.5032	1	9.5032	0.956	N.S.
ERROR(w)	119.302	12	9.9419	-	-
TIME	7.7175	2	3.8587	1.612	< 0.22
GVF x TIME	3.5911	2	1.7956	0.750	N.S.
MI x TIME	0.7550	2	0.3775	0.158	N.S.
GVF x MI x					
TIME	4.4496	2	2.2248	0.929	N.S.
ERROR(b)	57.4520	24	2.3938	-	-

TABLE 7
RADIAL POINTING ERROR ANOVA SUMMARY

Source	SS	df	MS	F	p
GVF	1.2746	1	1.2746	0.890	N.S.
MI	0.0012	1	0.0012	0.001	N.S.
GVF x MI	0.8523	1	0.8523	0.595	N.S.
ERROR(w)	17.1888	12	1.4324	-	-
TIME	2.3986	2	1.1993	28.214	< 0.001
GVF x TIME	0.0560	2	0.0280	0.658	N.S.
MI x TIME	0.1031	2	0.0516	1.213	< 0.35
GVF x MI x					
TIME	0.0348	2	0.0174	0.410	N.S.
ERROR(b)	1.0202	24	60.0425	-	-

TABLE 8
HORIZONTAL POINTING ERROR ANOVA SUMMARY

Source	SS	df	MS	F	p
GVF	4.8565	1	4.8565	3.597	< 0.08
MI	0.0303	1	0.0304	0.023	N.S.
GVF x MI	0.3974	1	0.3974	0.294	N.S.
ERROR(w)	16.2035	12	1.3503	-	-
TIME	0.7408	2	0.3704	4.594	< 0.025
GVF x TIME	0.1095	2	0.0547	0.679	N.S.
MI x TIME	0.1099	2	0.0550	0.682	N.S.
GVF x MI x TIME	0.0733	2	0.0367	0.455	N.S.
ERROR(b)	1.9351	24	0.0806	-	-

TABLE 9
VERTICAL POINTING ERROR ANOVA SUMMARY

Source	SS	df	MS	F	p
GVF	0.3431	1	0.3431	0.229	N.S.
MI	0.2221	1	0.2221	0.148	N.S.
GVF x MI	1.3492	1	1.3492	0.902	N.S.
ERROR(w)	17.9585	12	1.4965	-	-
TIME	1.9329	2	0.9664	33.405	< 0.001
GVF x TIME	0.0249	2	0.0124	0.430	N.S.
MI x TIME	0.1108	2	0.0554	1.915	< 0.20
GVF x MI x TIME	0.0417	2	0.0209	0.721	N.S.
ERROR(b)	0.6943	24	0.0289	-	-

The Pearson product-moment correlation between these values is 0.669. This is marginally significant ($0.05 < p < 0.10$, df = 7).

DISCUSSION

The results of this experiment are not consistent with the conflict hypothesis. In particular, it was expected that increased maneuvering intensity would result in increased sickness.

The results suggest that pilots are more likely to develop simulator sickness during flights at low altitudes. It would be appropriate to design training syllabi so that low level flights do not occur until the pilot has had time to adapt to the particular simulator. It would also be appropriate to consider increasing the altitude at which nauseogenic maneuvers are performed to the extent possible.

A positive relationship between the time flying the simulator and simulator sickness was evident. This suggests that the length of simulator sessions should be limited, at least until the pilot has adapted to the simulator.

ACKNOWLEDGEMENTS

REFERENCES

Henderson, B.W. (1989). Simulators Play Key Role In LHX Contractor Selection. Aviation Week & Space Technology. 27 November.

Kennedy, R.S, Dutton, B., Lilienthal, M.G., Ricard, G.L. & Frank, L.H. (1984). Simulator Sickness: Incidence of simulator aftereffects in 10 Navy flight trainers. Paper presented at the SAFE Symposium, Las Vegas, NV.

Kennedy, R.S., Lilienthal, M.G., Berbaum, K.S., Baltzley, D.R., & McCauley, M.E. (1989). Simulator Sickness in U.S. Navy Flight Simulators. Aviation, Space & Environmental Medicine. 60:10-16.

Lane, N.E., & Kennedy, R.S. (1988). A New Method for Quantifying Simulator Sickness: Development and Application of the Simulator Sickness Questionnaire (SSQ). Technical Report EOTR 88-7, Essex Corporation, Orlando, FL.

Lypaczewski, P.A., Jones, A.D., & Voorhees, J.W. (1986). Simulation of an Advanced Scout Attack Helicopter for Crew Station Studies. Proceedings of the Interservice/Industry Training Systems Conference, Nov. 18-20. p. 18-21.

Miller, J.C., Sharkey, T.J., Graham, G.A., & McCauley, M.E. (in preparation). Autonomic Physiological Data Associated with Simulator Discomfort.

Reason, J.T. & Brand, J.J. (1975). Motion Sickness. Academic Press, London.

Thomley, K.E., Kennedy, R.S., & Bittner, A.C. (1986). Development of Postural Equilibrium Tests for Examining Simulator Aftereffects. Perceptual Motor Skills, 63, 555-564.

Warren, R., Owen, D.H., & Hettinger, L.J. (1982). Separation of the Contributions of Optical Flow Rate and Edge Rate on the Perception of Egospeed Acceleration. In Owen, D. H. (1982). Optical Flow and Texture Variables Useful in Simulating Self Motion. Interim Technical Report 2/1/81 to 1/31/82, Air Force Office of Scientific Research, Bolling AFB, Washington D.C. AD-A117-016.

Author Index

(number after name indicates session in which paper appears)